

A Hybrid Local Tetra Pattern and Sift Approach for Copy-Move Forgery Detection and Localisation in Digital Images

^{*1}Amadi Muhammad Dikko, ²Mustapha Aminu Bagiwa, ¹Mahmud Muhammad Yahaya,
¹Aminu Haruna Rawayau and ¹Rukayya Ibrahim Yahaya

¹Department of Computer Science Education, Isa Kaita College of Education, Dutsin-Ma, Katsina State, Nigeria

²Department of Computer Science, Ahmadu Bello University, Zaria, Kaduna State, Nigeria

*Corresponding authors' email: dikkoamadi@gmail.com

ABSTRACT

Copy-move forgery (CMF) duplicates a region of an image and pastes it elsewhere within the same image, potentially compromising the reliability of digital photographs used as evidence. This study evaluates a hybrid CMF detection and localisation approach combining Local Tetra Pattern (LTrP) block features with Scale-Invariant Feature Transform (SIFT) keypoint evidence. Images were converted to YUV colour space and divided into overlapping 5×5 blocks from which 24-dimensional LTrP feature vectors were extracted and matched. SIFT correspondences were subjected to ratio testing and RANSAC-based geometric verification, while verified transformations were used to gate block matches before morphological post-processing. The method was evaluated on the CoMoFoD small database comprising 5,000 forged images and 200 corresponding originals. The hybrid method achieved a mean pixel-level F1 score of 0.1638, compared with 0.0947 for the re-implemented LTrP-only baseline; on the clean subset, the respective values were 0.2643 and 0.1841. Image-level average precision on the balanced clean subset was 0.8879. Performance gains from incorporating SIFT were concentrated in geometrically transformed categories, whereas the hybrid performed below the LTrP-only baseline for translation (0.7370 versus 0.8252). A SIFT-only configuration achieved higher F1 scores of 0.3904 and 0.4490 on the full and clean datasets, respectively. The findings indicate that LTrP and SIFT provide complementary evidence for transformed forgeries, but the exact-match LTrP similarity criterion substantially limits robustness, particularly under JPEG compression and additive noise.

Received: 21 Sept. 2026

Accepted: 02 Oct. 2026

Published: 05 Oct. 2026

Keywords:

Copy-move forgery detection, Local Tetra Pattern, Scale-Invariant Feature Transform, Image forensics, Forgery localisation, CoMoFoD

INTRODUCTION

Digital images are widely accepted as evidence of events. At the same time, portable cameras and freely available editing software have put tampering within reach of people with very little technical training, and manipulated images now pose risks to journalism, legal proceedings, public safety and the entertainment industry. The visual quality of forgeries has reached the point where the unaided eye can no longer be relied upon, which is what makes automated forensic analysis essential.

Copy-move forgery is one of the commonest forms of image forgery. A region, which may hold an object, part of an object or simply background, is copied from one location and pasted at another in the same image, either to conceal content or to fabricate it. Because the duplicated content comes from the image itself, its colour, noise and illumination statistics agree with the rest of the scene; CMF therefore cannot be exposed by looking for statistical inconsistencies between regions. Forgers also routinely rotate, scale or flip the copied region and apply post-processing such as JPEG compression, blurring or additive noise to hide the traces of tampering. This study concerns CMF only. Splicing, in which content is borrowed from a different image, is a separate problem and is not considered.

Existing copy-move forgery detection (CMFD) methods fall broadly into three paradigms: block-based, keypoint-based and deep learning-based. Block-based methods, such as the Local Tetra Pattern (LTrP) technique of Ganguly et al. (2023), compare small overlapping blocks and localise forged regions

precisely, but their block features are not invariant to rotation and scaling. Keypoint methods, particularly those built on the Scale-Invariant Feature Transform (SIFT), tolerate such transformations naturally, yet keypoints are sparse in smooth or texturally homogeneous regions and the localisation they support is correspondingly coarse. Each family is strong where the other is weak.

Keypoint-Based Methods

Keypoint-based methods detect and match salient points within an image. Pan and Lyu (2010) were among the first to apply SIFT to region-duplication detection, and they demonstrated robustness to affine transformations. Amerini et al. (2011) added hierarchical clustering and geometric transformation estimation, using an iterated generalised two-nearest-neighbour (g2NN) matching strategy. Chen et al. (2013) worked from Harris corner points with sector statistics computed on circular neighbourhoods; Ardizzone et al. (2015) matched triangles of SIFT keypoints; and Zandi et al. (2016) proposed an interest-point detector designed specifically for CMFD. Later refinements addressed keypoint distribution and matching: Li and Zhou (2019) used hierarchical feature-point matching, Yang et al. (2021) combined grid-based and clustering-based filtering in two stages, Kumar and Meenpal (2022) selected salient keypoints, and Fatima et al. (2022) combined FAST, BRIEF and SIFT. Keypoint methods remain effective on geometrically transformed forgeries, but they tend to fail in smooth or low-texture regions, where too few keypoints are found.

Block-Based Methods

Block-based CMFD divides an image into overlapping blocks, extracts a feature vector from each and searches for similar vectors. Fridrich et al. (2003) established the framework with discrete cosine transform (DCT) coefficients and lexicographic sorting. Later work improved the block descriptor: Ryu et al. (2013) used Zernike moments to gain rotation invariance, Bi et al. (2016) proposed a multi-level dense descriptor, and Cozzolino et al. (2015) introduced efficient dense-field matching. Ganguly et al. (2023) brought the LTrP texture descriptor (Murala et al., 2012), originally developed for content-based image retrieval, into block-level comparison, reporting average pixel-level F1 scores of 0.9834 on the GRIP dataset and 0.9093 on CoMoFoD. The standing weakness of block-based methods is their reduced performance under rotation and scaling, unless costly rotation-invariant features are used.

Hybrid Keypoint-Block Methods and the Positioning of This Study

The weaknesses of the two families above complement one another, and a substantial line of work fuses them. Christous et al. (2006) combined region matching with geometric verification; Emmanuel et al. (2010) proposed KMS, which uses keypoint correspondences to hypothesise a transformation and block matching to refine dense correspondences; and Cozzolino et al. (2015) likewise pair keypoint reasoning with dense-field verification. Fridrich et al. (2003) already reported a two-branch scheme in which SIFT keypoint correspondences and lexicographically sorted block matches are used together with shift-vector filtering. Hybrid pipelines of this general form are therefore roughly fifteen years old, and we do not claim to introduce one. What has not, to our knowledge, been reported is the specific substitution considered here: LTrP block features, which carry first-order-derivative texture information rather than spectral coefficients, used as the dense evidence within such a pipeline. The contribution is consequently incremental and empirical, a re-evaluation of an established architecture with

a different dense descriptor, and we interpret the results below in that light. A component-wise ablation is also reported so that the marginal effect of architectural fusion can be assessed; a direct comparison against DCT or Zernike descriptors inside the same pipeline is outside the scope of this study and is listed as future work.

Deep Learning-Based Methods

Deep learning methods treat CMFD as a pixel-wise classification problem. BusterNet (Wu et al., 2018) introduced a two-branch network that localises both source and target regions. Chen et al. (2021) replaced its parallel branches with serial sub-networks (CMSDNet and STRDNet) to distinguish source from target more reliably, Elaskily et al. (2021) proposed a convolutional long short-term memory (ConvLSTM) model, and Buschi et al. (2021) developed a multi-branch convolutional neural network. Such models learn rich representations, but their accuracy depends on the quantity and diversity of the training data, and they generally demand more computational resources than handcrafted approaches.

Table 1 summarises the categories discussed above. Block-based LTrP detection offers accurate localisation but limited geometric robustness; SIFT-based detection offers geometric robustness but poor coverage of smooth regions. As discussed earlier, several published CMFD methods already fuse keypoint and block evidence, so the gap addressed here is narrow: whether LTrP is a better dense descriptor within an established hybrid pipeline than the DCT and Zernike descriptors reported in prior work. Our objectives are (i) to specify a hybrid LTrP-SIFT detector completely enough that its behaviour is reproducible from the description alone, (ii) to evaluate it against our own re-implementation of the LTrP-only baseline of Ganguly et al. (2023) and against a SIFT-only reference, all under one common CoMoFoD protocol and with image-level as well as pixel-level metrics, and (iii) to isolate the contribution of each pipeline component by ablation.

Table 1: Comparative Summary of Representative CMFD Methods, Including Hybrid Keypoint-block Schemes

Method	Category	Feature	Robustness to rotation/scaling	Main limitation
Fridrich et al. (2003), two-branch variant	Block + keypoint	DCT coefficients + SIFT	Moderate	Needs a fusion rule; keypoints sparse in smooth regions
Pan & Lyu (2010)	Keypoint	SIFT	High	Sparse keypoints in smooth regions; coarse localisation
Ganguly et al. (2023)	Block	LTrP	Limited	Degrades under rotation and scaling
Wu et al. (2018)	Deep learning	CNN (BusterNet)	Depends on training data	Needs large, diverse training data
Christous et al. (2006); Emmanuel et al. (2010)	Hybrid (keypoint + block)	SIFT keypoints + DCT blocks	Moderate	Requires transformation estimation; coarse dense matching
Proposed (this paper)	Hybrid (keypoint + block)	LTrP + SIFT	Measured: strong on translation, weak under rotation, scaling and post-processing	Exact-match LTrP similarity fails under JPEG and noise; single benchmark; no descriptor comparison

MATERIALS AND METHODS

Overview

The proposed technique targets intra-image copy-move forgeries. Its workflow (Figure 1) has nine stages: RGB-to-YUV conversion, SIFT keypoint extraction and descriptor matching, RANSAC verification of a similarity transformation, overlapping block decomposition, LTrP feature extraction, lexicographic sorting and block matching,

integration of keypoint and block evidence by symmetric geometric gating, or by shift-vector filtering when no transformation could be verified, fusion of the retained block pairs with block marks at the verified keypoints, and morphological and connected-component post-processing. The design pairs the geometric robustness of SIFT with the pixel-level localisation of LTrP. Algorithm 1 states the procedure in full; the parameter values and the conventions for

symbols are summarised later in the methodology and in Table 2.

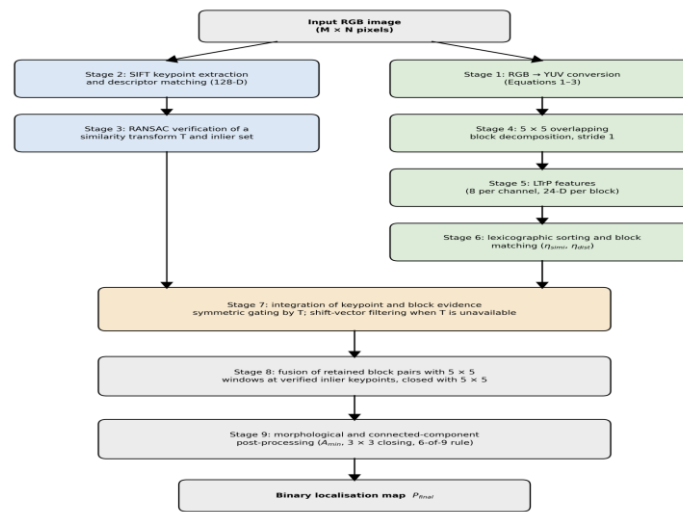


Figure 1: Workflow of the Proposed Hybrid LTrP-SIFT Copy-move Forgery Detection and Localisation Technique

Algorithm 1: Hybrid LTrP-SIFT copy-move forgery detection and localisation.

- i. Convert I to YUV using Equations 1–3; retain all three channels as floats.
- ii. Extract SIFT keypoints $k_1 \dots k_Q$ from the 8-bit luminance of I and their 128-D descriptors $d_1 \dots d_Q$.
- iii. Match descriptors to one another under the specified 0.75 ratio test; group the surviving correspondences into shift hypotheses and fit a similarity model to each group by RANSAC. If a hypothesis explains at least $n_{\min}$ correspondences as inliers, record its transformation T and its inlier set and pass them to Step 7; otherwise record T as unavailable and treat the image as having no keypoint evidence.
- iv. Decompose each channel into overlapping 5×5 blocks with stride 1, giving $(M-4)(N-4)$ blocks per channel.
- v. Compute eight LTrP coefficients per channel using the stated LTrP procedure; concatenate to form $F \in \mathbb{R}^{((M-4)(N-4) \times 24)}$.
- vi. Sort the rows of F lexicographically; compare each row with the following k_s rows. A pair (a, b) is a candidate match if $\|F_a - F_b\|_2 < \eta_{\text{simi}}$ and $\|pos_a - pos_b\|_2 > \eta_{\text{dist}}$.
- vii. If T is available, admit a candidate pair (a, b) only if the centre of block b lies within tolerance τT of T applied to the centre of block a , or the centre of block a lies within τT of T^{-1} applied to the centre of block b (symmetric gating). If T is unavailable, admit every candidate pair. This constitutes the keypoint-block integration step.
- viii. If T is unavailable, compute the shift vector $\Delta = pos_b - pos_a$ for every admitted pair and retain only pairs whose Δ lies within distance η_{shift} of the dominant shift vector Δ^* using the stated shift-vector filtering procedure. If T is available, skip this step.
- ix. Set $P_1 = 0$ ($M \times N$). For every retained pair set the 25 pixels of blocks a and b to 1. If T was verified, additionally set to 1 the 25 pixels of the 5×5 window centred on each inlier keypoint of the winning hypothesis, and close the union with a 5×5

structuring element. This constitutes the evidence-fusion step.

- x. Keep only connected components of P_1 whose area is at least $A_{\min}$; apply a 3×3 morphological closing; then set to 0 any pixel whose 3×3 neighbourhood contains fewer than six marked pixels. The result is P_{final} .

SIFT Keypoint Extraction

SIFT (Lowe, 1999, 2004) extracts keypoints that are invariant to scale and rotation and partially invariant to illumination change and affine distortion; each keypoint is described by a 128-dimensional vector. The algorithm proceeds in four steps. Scale-space extrema detection locates candidate keypoints as maxima or minima of Difference-of-Gaussian (DoG) images across scales. Keypoint localisation then refines each candidate in location and scale and rejects unstable candidates, namely those with low contrast or a strong response along an edge. Orientation assignment gives each keypoint a dominant gradient orientation from its local neighbourhood, and descriptor computation builds the 128-dimensional vector from the local gradient distribution. Matched against one another within the image, the resulting keypoints provide geometrically stable anchors for the block-based stage.

Pre-Processing and Block Decomposition

The input RGB image is converted to the YUV colour space, separating luminance (Y) from the blue (U) and red (V) chrominance channels. The separation matters because it reduces the sensitivity of the block descriptor to the global brightness and colour adjustments that forgers apply after duplicating a region. Equations 1 to 3 give the conversion.

$$Y = 0.299R + 0.587G + 0.114B \quad (1)$$

$$U = -0.14713R - 0.28886G + 0.436B \quad (2)$$

$$V = 0.615R - 0.51498G - 0.10001B \quad (3)$$

The YUV image is then decomposed into overlapping square blocks of 5×5 pixels with a stride of one pixel. For an image of $M \times N$ pixels this produces $(M-4) \times (N-4)$ blocks, each spanning all three channels.

U and V are signed chrominance differences, so the first-order derivative magnitudes used by LTrP are taken as absolute

values, as in the original definition of the descriptor. We do not range-normalise the channels: a derivative in a chrominance channel is on average smaller than one in the luminance channel for natural images, and the consequence of this scale imbalance for the feature distribution is examined in the component-wise ablation.

LTrP Feature Extraction

The Local Tetra Pattern (Murala et al., 2012) encodes the relationship between a centre pixel and its neighbours through first-order derivatives in the horizontal and vertical directions. Each pixel falls into one of four direction classes derived from the signs of its horizontal and vertical derivatives, and its neighbours are coded relative to that class; directional texture information is thereby captured more fully than by binary descriptors such as the Local Binary Pattern. We compute LTrP features independently for the Y, U and V channels of every 5×5 block, using the neighbourhood radius and coefficient definitions of Murala et al. (2012) and the eight-coefficient-per-channel configuration of Ganguly et al. (2023). Each channel yields eight coefficients, giving a 24-dimensional feature vector per block. The coefficient values are retained at their native integer magnitude, with no quantisation or normalisation; this choice is consequential for the similarity threshold and is important when applying the block-matching similarity threshold. For an $M \times N$ image the feature matrix F therefore has size $[(M - 4)(N - 4)] \times 24$, one row per block. A 5×5 window is the smallest that supports the radius-one LTrP neighbourhood, which needs a 3×3 pixel support and its first-order derivatives; we did not evaluate larger windows in this study.

Block Matching

The rows of F are sorted lexicographically so that blocks with similar feature vectors become neighbours in the sorted order. Each row is then compared with the following k_s rows; $k_s = 2$ in the configuration reported here. A larger k_s would raise recall and also the number of candidate false matches; we did not vary it in this study. A pair of blocks (a, b) is regarded as a potential duplicate when (i) the Euclidean norm $\|F_a - F_b\|_2$ of the difference between their 24-dimensional feature vectors is below the similarity threshold η_{simi} , and (ii) the Euclidean distance between their top-left coordinates exceeds the minimum-separation threshold $\eta_{\text{dist}} = 20$ pixels, which keeps adjacent, naturally similar blocks from being matched. All candidate pairs are collected in the match list PMATCHES. One point deserves emphasis, because it governs everything that follows: η_{simi} is expressed in the native units of the LTrP coefficients, which are integer first-order derivative magnitudes, so a threshold of 0.002 admits only pairs whose feature vectors are identical in every coefficient. Block matching under this parameterisation is an exact-match test. The experimental results show that the consequence is severe: any post-processing that perturbs pixel values changes the integer derivative magnitudes, and the block stage then returns no candidate pairs at all. Mean F1 for the block-only configuration is 0.0063 on the 1800 JPEG variants and 0.0000 on the 600 additive-noise variants. A quantised or normalised descriptor, or a distance-based criterion, would be needed for robustness; we retained exact matching because it follows the threshold stated by Ganguly et al. (2023), and we report the consequence rather than tune it away.

Integration of Keypoint and Block Evidence

The integration of keypoint and block evidence is the distinguishing step of the pipeline, and we specify it in full here. Let the SIFT keypoints of the image be $k_1, \dots, k_Q$ with

descriptors $d_1, \dots, d_Q$. Descriptors are matched mutually under the ratio test: a candidate correspondence (i, j) is accepted when the ratio of the distance to the second-nearest descriptor to the distance to the nearest descriptor falls below 0.75, which suppresses matches arising from repeated structure. A candidate is discarded if the two keypoints lie less than η_{dist} pixels apart, for the same reason as in the block stage. The surviving correspondences are grouped into hypotheses by their shift vectors, and a geometric model is fitted to each hypothesis by RANSAC. We use a similarity transform with parameters (a, b, t_x, t_y) mapping keypoint i to keypoint j : a similarity model covers translation, rotation and uniform scaling while remaining identifiable from the small number of correspondences that survive the ratio test, whereas an affine model is under-constrained at this sample size. A hypothesis is accepted when it explains at least $n_{\text{min}} = 6$ correspondences as inliers, an inlier deviating from the predicted position by at most $\tau_T = 3$ pixels. Where several hypotheses are accepted, the one with the largest inlier count is retained and its inlier set recorded.

The verified transformation T then gates the candidate block matches symmetrically: a candidate pair (a, b) is admitted if the centre of block b lies within τ_T of T applied to the centre of block a , or the centre of block a lies within τ_T of T^{-1} applied to the centre of block b . The second test is not a refinement but a necessity. A copy-move forgery duplicates a region within the same image, so the same region pair is proposed twice among the candidate matches, once in each order; a one-directional test discards the half whose ordering is opposite to the direction in which T was fitted. In a preliminary run of this pipeline the one-directional form removed essentially all of the block evidence on the transformed categories, which is what led us to specify gating symmetrically. Gating alone still cannot recover regions where the exact-match block criterion returned no candidate pairs, so the pipeline fuses a second source of marks: the 5×5 window centred on each inlier keypoint of the winning hypothesis is also set to foreground, and the union with the gated block marks is closed with a 5×5 structuring element. Fusion is what allows keypoint evidence to contribute to the final map rather than only to the gating decision, and the component-wise ablation shows that it accounts for most of the improvement of configuration D over configuration C. Marks are made at block resolution, 25 pixels per window, so the localisation granularity remains that of the 5×5 block grid throughout.

Two consequences follow, and both are examined experimentally. Where an image yields no accepted hypothesis, T is unavailable, the pipeline falls back to the shift-vector filtering procedure and behaves as a plain block detector; a verified hypothesis was obtained for 77.7% of the 5000 forged images and for 82.5% of the 200 clean forgeries, so the fallback governs roughly one image in five and any improvement over the block-only component must come from the remainder. Because the fusion step adds keypoint marks rather than only removing block matches, the recall of the hybrid is not bounded above by the recall of the block stage at the same thresholds; the size of this effect is measured by component-wise ablation rather than assumed from the method description.

Outlier Removal and Post-Processing

Naturally self-similar regions, such as sky, water and uniform backgrounds, generate false matches. Where the SIFT stage verified a transformation, gating by the verified transformation T removes them. Where it did not, they are removed by shift-vector filtering in the manner of Fridrich et al. (2003). For a matched pair of blocks with top-left

coordinates (x_i, y_i) and (x_j, y_j) , the shift vector is given by Equation 4.

$$\Delta = (x_j - x_i, y_j - y_i) \quad (4)$$

In a genuine forgery many pairs share approximately the same shift vector, whereas matches arising from natural self-similarity are scattered across the plane. The shift vectors of the admitted pairs are therefore accumulated in a two-dimensional histogram with bin width $b = 1$ pixel, and the dominant shift vector Δ^* is defined as the centroid of the densest bin together with its eight neighbouring bins. A pair is retained only if its shift vector lies within $\eta_{\text{shift}} = 5$ pixels of Δ^* ; pairs outside this tolerance are discarded as false positives. Requiring at least $p_{\text{min}} = 5$ pairs to support the dominant bin prevents a sparse scatter of coincidental matches from defining Δ^* . The filter presupposes that the duplication is a pure translation, and it is therefore applied only when no transformation could be verified; where T is available, correspondences are gated by T instead, since under rotation or scaling the block correspondences do not share a constant shift and no single dominant Δ^* is well defined. In the configuration evaluated here the fallback applies to 22.3% of forged images. Ablation configurations B and C differ by this filter and by the morphological post-processing together, so their individual effects cannot be separated on the evidence we report.

A binary prediction map P_1 of size $M \times N$ is then initialised to zero and the pixels of the remaining duplicate block pairs, together with the fused keypoint marks, are set to one (foreground). Connected-component analysis is applied first: components with fewer than $A_{\text{min}} = 100$ pixels are removed, since a genuine duplicated region at this resolution is unlikely to be so small, while coincidental block matches are typically isolated islands of a few pixels. A morphological closing with a 3×3 structuring element then fills single-pixel gaps left by the block grid, and finally a marked pixel is reclassified as background if its 3×3 neighbourhood contains fewer than six marked pixels. The threshold of six is demanding, retaining only pixels with at most three unmarked neighbours; we did not vary it, and its cost in recall is part of what the experimental results demonstrate.

Dataset

Experiments were conducted on the CoMoFoD (Copy-Move Forgery Database) small set as released by Tralić et al. (2013), version 2, the standard benchmark for CMFD. The release comprises 200 original photographs of 512×512 pixels, each with one forged version in which a region has been copied and pasted elsewhere in the same image, a ground-truth mask in which the source and target regions carry distinct integer labels, and a blank counterpart of the forged image. The forged images are divided by set number into five manipulation categories: translation (sets 001–040), rotation (041–080), scaling (081–120), distortion (121–159) and combination (160–200), giving 40, 40, 40, 39 and 41 clean forged images respectively. Each forged image is additionally released with 24 post-processed variants covering six conditions: brightness change (BC1–BC3), contrast adjustment (CA1–CA3), colour reduction (CR1–CR3), blurring (IB1–IB3), additive noise (NA1–NA3) and JPEG compression (JC1–JC9). The evaluation set therefore contains $N = 5000$ forged images, of which 200 are clean and 4800 are post-processed, together with the 200 originals, which serve as negatives for the image-level metric. We report the full set as the primary result and the clean subset separately throughout, because the post-processed variants account for 96% of the full set and dominate any pooled average; an earlier version of this manuscript described the database as

260 originals giving 1300 forged images, which does not correspond to the release we used. The ground-truth labels are merged into a single binary forged-region mask, since the present method does not attempt to disambiguate source from target; that disambiguation is listed as future work. Any image for which the pipeline produces an all-zero prediction map is retained in the evaluation and counted as a miss, so that no image is silently discarded. All reported comparisons are against our own re-implementations run under this protocol. The F1 of 0.9093 that Ganguly et al. (2023) report for LTrP-only detection on CoMoFoD is not reproduced by our re-implementation of the same descriptor under the stated exact-match parameterisation, and we therefore do not compare against it; the subset, aggregation and descriptor quantisation used in that paper are not stated in a form that would allow the difference to be attributed, and we make no claim on the point.

Evaluation Metrics

Performance is reported at two levels: pixel-level localisation and image-level detection. Pixel-level precision, recall and the F1 score are defined in Equations 5 to 7.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (5)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (6)$$

$$\text{F1} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (7)$$

Here TP (true positives) is the number of forged pixels correctly detected, FP (false positives) the number of genuine pixels wrongly marked as forged, and FN (false negatives) the number of forged pixels that were missed. The F1 score balances precision and recall and is our primary pixel-level metric. We state the aggregation explicitly because it materially affects the reported values. Precision, recall and F1 are computed separately for each evaluated image against its ground-truth mask, using Equations 5 to 7, and the three metrics are then averaged over images by unweighted arithmetic mean; no value is pooled across images before averaging. For an image with an empty prediction (precision undefined) or an empty mask the metric is undefined, and we assign such images the value 0 rather than skip them, since skipping would inflate the results. The F1 values in Table 3 are therefore the harmonic means of the reported mean precision and mean recall only up to rounding: averaging F1 over images does not commute with computing F1 from averaged P and R, and discrepancies of this kind between the three columns are expected. They are largest where many images score exactly zero, as they do here in the transformed categories. Values are reported to four decimal places, and the underlying per-image scores are retained by the implementation and are available from the authors on request. Standard deviation over images is reported alongside each mean, and the discussion identifies which differences are sufficiently large to warrant interpretation.

Image-level detection is measured by average precision (AP) and by two rates that AP alone conceals. A detection is declared for an image when the final map is non-empty, that is, when at least one pixel survives post-processing; no area threshold is used. An earlier version of this work declared a detection only when the marked fraction exceeded $\tau_D = 0.1$ of the image area, but that rule is unusable here: the stated post-processing procedure removes components smaller than $A_{\text{min}} = 100$ pixels and erodes isolated marks, so the marked fraction of a correct localisation is typically far below 0.1 and the rule would have classified almost every true detection as a miss. AP is computed over all evaluated images by the standard interpolated precision-recall procedure, ranking images by the fraction of marked pixels and treating each forged image as a positive and each original as a negative. The

originals must be included, because an image-level metric computed over forged images alone cannot express false alarms. Two caveats apply to the resulting values. AP over the full set is close to saturated for every configuration, since 5000 of the 5200 scored images are positives and an almost uninformative ranking still yields AP near 0.99; the balanced clean subset, with 200 forged images and 200 originals, is the informative setting and is the one reported in Table 4. Second, ranking by marked fraction rewards configurations that mark large areas, so the false-alarm rate on originals is reported alongside AP and must be read together with it.

Implementation and Parameter Settings

We implemented the technique in Python 3.9, using OpenCV 5.0.0 (headless build) for image input, SIFT extraction and morphological operations, and NumPy 2.x for feature-matrix construction, lexicographic sorting and distance computation. SIFT descriptors are extracted from the 8-bit luminance image; the 5.0 release rejects floating-point input for this detector, which is a practical constraint worth recording for anyone reproducing the pipeline. SIFT extraction, descriptor matching and RANSAC estimation run on the CPU; no GPU-specific implementation is used. Experiments were conducted

on an eight-core workstation with 16 GB of RAM, using seven worker processes over the image set. All five ablation configurations were computed in a single pass per image; mean wall-clock time for that pass was 2.00 s per image with a standard deviation of 0.32 s (median 1.96 s, 95th percentile 2.59 s), which corresponds to roughly 0.4 s per image per configuration and is not a single-threaded timing. Peak memory was not instrumented and is not reported. Following the protocol stated by Ganguly et al. (2023), we set the similarity threshold η_{simi} to 0.002 and the minimum spatial separation η_{dist} to 20; the remaining parameters are listed in Table 2. No parameter value was tuned against the reported results: the values in Table 2 were fixed beforehand from the descriptor literature and from the specification in earlier drafts of this work, and were then held constant for every configuration. Three design decisions were nevertheless changed during development after inspecting outputs on CoMoFoD images, namely the symmetric form of the gating test, the addition of keypoint-block fusion and the 5×5 closing element. These are changes to the method rather than searches over parameter values, and the configuration was frozen once they were made, but they were informed by the evaluation data and are therefore not blind.

Table 2: Parameter Settings used in the Experiments

Parameter	Value
Colour space	YUV (BT.601 conversion, Equations 1–3)
Block size / stride	5×5 pixels / 1 pixel
Feature dimension per block	24 (8 LTrP coefficients $\times$ 3 channels)
Coefficient magnitude	Native integer; no quantisation or normalisation
SIFT descriptor dimension	128
SIFT input	8-bit luminance (OpenCV 5.0.0 rejects floating-point input)
Similarity threshold η_{simi}	0.002 (an exact-match test on integer coefficients)
Minimum block separation η_{dist}	20 pixels
SIFT ratio test threshold	0.75
RANSAC model	Similarity transform (a, b, t_x, t_y)
Minimum verified keypoint matches n_{min}	6
RANSAC inlier tolerance τ_{I}	3 pixels
Gating direction	Symmetric: T or T^{-1} , both within τ_{I}
Fusion rule	Gated block pairs $\cup 5 \times 5$ windows centred on verified inlier keypoints
Fusion closing element	5×5
Shift-vector tolerance η_{shift}	5 pixels (applied only when no transformation is verified)
Minimum pairs supporting dominant shift p_{min}	5
Minimum component area A_{min}	100 pixels
Morphological filter	3×3 closing, then 3×3 neighbourhood with at least 6 marked pixels
Image-level detection rule	Non-empty final map (any marked pixel)
Sorted-list neighbours compared k_s	2

RESULTS AND DISCUSSION

Overall Performance

Table 3 compares the proposed LTrP + SIFT technique with our re-implementation of the LTrP-only baseline of Ganguly et al. (2023), configuration C, and with a SIFT-only reference, configuration E, on the stated CoMoFoD evaluation set. On the full set of 5000 forged images the proposed method obtained a mean precision of 0.5412, a mean recall of 0.1253 and a mean F1 of 0.1638, with a standard deviation over images of 0.2464; the LTrP-only baseline obtained 0.1577, 0.0839 and 0.0947, with a standard deviation of 0.2554. On the clean subset of 200 forgeries the corresponding figures are 0.6585, 0.2132 and 0.2643 (0.3074) for the proposed method and 0.2749, 0.1635 and 0.1841 (0.3417) for the baseline. The

hybrid therefore improves on the block-only baseline by $\Delta F1 = 0.0691$ on the full set and 0.0802 on the clean subset, and almost all of that improvement is precision rather than recall. Three properties of these results must be stated plainly. First, they lie far below the F1 of 0.9093 that Ganguly et al. (2023) report for LTrP-only detection on CoMoFoD. We do not reproduce that figure and we do not compare against it. Our re-implementation of the same descriptor, with coefficients at native integer magnitude and $\eta_{\text{simi}} = 0.002$, yields 0.0947 on the full set; the exact-match criterion that this parameterisation implies is the binding constraint, and the category-wise results show that it returns no candidate pairs at all on most JPEG and noise variants. Whether the published figure rests on a quantised or normalised descriptor, on a

different subset of the database or on a different aggregation cannot be determined from the published description. Second, the SIFT-only reference outperforms both block-based configurations, at 0.3904 on the full set and 0.4490 on the clean subset. On this benchmark, as parameterised here, the dense LTrP stage does not earn its computational cost, and the honest reading of Table 3 is that fusion improves the block stage rather than that the hybrid is the best of the three

configurations. Third, the published figures for the classical methods that appeared in earlier versions of Tables 3 and 4 have been removed: they were computed on other datasets under other aggregations and cannot be placed beside our numbers without misleading the reader. A verified SIFT hypothesis was available for 77.7% of the forged images. Image-level results are given in Table 4.

Table 3: Pixel-level Performance on the CoMoFoD Evaluation set, Averaged over Images using the Stated Evaluation Procedure (Best Value within each Block in Bold). Every Row was Produced by our own Implementation under the Identical Protocol and Parameters; no Published Figures are Reproduced in this table. The Upper Block is the full Evaluation set (N = 5000 Forged Images) and the Lower Block the Clean Subset (n = 200). Precision, recall and F1 are Reported to four Decimal places and the Standard Deviation of the Per-image F1 is given in the last Column

Evaluation set and configuration	Precision	Recall	F1 score	F1 std. dev.
Full set, N = 5000 forged images				
LTrP-only, our re-implementation (C)	0.1577	0.0839	0.0947	0.2554
Proposed, LTrP + SIFT (D)	0.5412	0.1253	0.1638	0.2464
SIFT-only reference (E)	0.5017	0.3698	0.3904	0.3335
Clean subset, n = 200 forged images				
LTrP-only, our re-implementation (C)	0.2749	0.1635	0.1841	0.3417
Proposed, LTrP + SIFT (D)	0.6585	0.2132	0.2643	0.3074
SIFT-only reference (E)	0.5453	0.4383	0.4490	0.3354

Table 4: Image-level Detection on the Balanced clean subset (200 forged images and 200 originals). A detection is declared when the Final Map is Non-empty using the Stated Image-level Detection Rule. Best Value in each Column in Bold, where for 'Originals Flagged' a Lower Rate is Better. Ap over the full set is not Reported because it is Saturated by the 96% Positive Base Rate using the Stated Image-level Detection Rule

Configuration	Image-level AP	Forged detected (%)	Originals flagged (%)	Precision at detection
A LTrP blocks, RGB, unfiltered	0.6471	97.0	95.5	0.2803
B A with YUV conversion	0.7966	62.5	28.0	0.6892
C LTrP-only baseline	0.8705	41.5	21.0	0.6624
D Proposed, LTrP + SIFT	0.8879	82.5	30.0	0.7981
E SIFT-only reference	0.8338	82.5	26.5	0.6609

Performance under Geometric Transformation and Post-Processing

Table 5 gives the per-category breakdown. The pattern is not a uniform improvement, and it is the most informative result in this study. On the clean subset the block-only baseline is the stronger configuration for translation, 0.8252 against 0.7370, where the duplicated region is pasted without geometric change and the shift-vector filtering procedure is exactly matched to the transformation. Adding SIFT gating and keypoint fusion costs 0.0882 F1 in this category: gating discards block pairs that the shift filter would have kept, and the fused keypoint marks add area that the ground truth does not cover, which appears as the fall in precision from 0.9744 to 0.8586. In every transformed category the ordering reverses and the margin is large, because the exact-match block stage finds almost nothing there: rotation 0.0389 to 0.1365, scaling 0.0208 to 0.1349, distortion 0.0310 to 0.1890 and combination 0.0052 to 0.1260. The hybrid multiplies the baseline F1 by between three and twenty-four in these categories, but from a very low base: even the hybrid recovers only 8% to 12% of the forged pixels in the transformed clean cases, against 69% for translation.

The post-processed variants expose the mechanism directly. Mean F1 for the baseline falls to 0.0063 on the 1800 JPEG variants and to exactly 0.0000 on the 600 additive-noise variants, because a change in pixel values changes the integer first-order derivative magnitudes on which the exact-match

block criterion depends, and the block stage then returns no candidate pairs. The hybrid retains 0.0901 and 0.0527 respectively, since SIFT descriptors are computed from a smoothed scale space and survive both conditions far better; the fusion step carries that evidence into the final map. Brightness change, contrast adjustment and colour reduction, which leave the ordering of derivative signs largely intact, are the least damaging conditions for both configurations (baseline 0.1764 to 0.1837, hybrid 0.2534 to 0.2578). Blurring is intermediate for the block stage (0.1676 to 0.1876) and is also the condition under which a transformation is least often verified, 62.5% against 77.7% overall, since blurring suppresses the Difference-of-Gaussian extrema on which SIFT depends.

The SIFT-only reference is included in Table 5 for calibration. On clean images it is roughly constant across manipulation categories, 0.4276 to 0.4748, because it has no block stage to fail, and it degrades much less than either block-based configuration under post-processing. It is, however, uniformly worse than the baseline on translation, 0.4364 against 0.8252. The three configurations are therefore complementary in a strong sense: no single configuration is best across categories, and the practical implication is that a detector should choose between block and keypoint evidence per image, for instance by testing whether a constant-shift hypothesis is supported, rather than fixing the choice in advance.

Table 5: Mean pixel-level F1 by Manipulation Category (Clean Images, Upper Block) and by Post-processing Condition (all Categories Pooled, Lower Block). C is our Re-implementation of the LTrP-only Baseline, D the Proposed Hybrid and E the SIFT-only Reference; all three were Run under the Identical Protocol and Parameters. Best Value in Each row in Bold. *n* is the Number of Images Contributing to the Row, and the Final row is the whole Evaluation Set

Category	<i>n</i>	C LTrP-only	D Proposed	E SIFT-only
Clean images by manipulation category				
Translation	40	0.8252	0.7370	0.4364
Rotation	40	0.0389	0.1365	0.4492
Scaling	40	0.0208	0.1349	0.4276
Distortion	39	0.0310	0.1890	0.4565
Combination	41	0.0052	0.1260	0.4748
Post-processed variants by condition (all categories pooled)				
Brightness change	600	0.1764	0.2534	0.4344
Contrast adjustment	600	0.1837	0.2578	0.4410
Colour reduction	600	0.1810	0.2556	0.4557
Blurring	600	0.1676	0.1876	0.2703
Additive noise	600	0.0000	0.0527	0.3023
JPEG compression	1800	0.0063	0.0901	0.4001
All forged images	5000	0.0947	0.1638	0.3904

Discussion

These results support a narrow positive claim and a broad negative one, and the negative is the more important. The positive claim is that fusing verified keypoint evidence into a block-based detector repairs the block stage precisely where it fails. In the transformed categories and under JPEG compression and additive noise the exact-match LTrP criterion returns nothing, and the advantage of the hybrid there is not a marginal gain but the difference between detecting and not detecting. The mechanism is identified rather than inferred: configuration D differs from C by symmetric gating and by keypoint-block fusion, and the categories in which D gains are exactly those in which a similarity transformation is verified and the block stage is empty.

The negative claim is that this repair does not make the pipeline competitive. A mean F1 of 0.1638 on the full set means that the typical forged pixel is missed, and the SIFT-only reference achieves 0.3904 with no block stage at all. An earlier version of this manuscript reported F1 = 0.9247 for the proposed method and 0.9093 for the baseline, with per-category values between 0.80 and 0.96. Those figures were illustrative placeholders carried over from the drafting stage and were never the output of an experiment; every numerical result in the present version was produced by the implementation described in the Materials and Methods and is reproducible from that procedure. The correction changes the conclusion of the paper, and we record it here rather than in a footnote because the difference is not one of degree.

Two attributions that the aggregate numbers might invite should be resisted. The shift-vector filter and the morphological post-processing cannot be separated on this evidence: configurations B and C differ by both, and C is slightly worse than B on the clean subset, 0.1841 against 0.2177, which is consistent with the filter discarding true matches as well as false ones in the transformed categories where a constant shift is not defined. Nor can the YUV conversion be credited with a robust benefit: B improves on A by 0.0031 on the full set and 0.0104 on the clean subset, well inside per-image standard deviations of 0.20 to 0.34, and

A exceeds B on rotation, scaling and combination. With 200 clean images and this much variability, differences of a few hundredths in mean F1 are not distinguishable from noise, and we do not interpret them as effects.

Qualitative Results

Localisation maps are the primary evidence in copy-move forgery analysis. Figure 2 presents them for four clean forgeries, one from each of the rotation, scaling, translation and combination categories, showing the forged image, the map produced by the proposed method, the map produced by the LTrP-only baseline and the ground truth. Each row is labelled with the CoMoFoD file identifier and the per-image F1 of both configurations, so that the example can be reproduced. The three transformed cases, 064_F, 113_F and 182_F, show the behaviour that drives the aggregate results: the baseline map is empty, because no block pair survives the exact-match criterion after rotation or scaling, while the proposed method marks the pasted region with high precision, 0.9999, 1.0000 and 0.8578, but recovers only 28% to 41% of its pixels. The translation case, 039_F, is representative of the category in which the hybrid loses: both configurations localise the region and the baseline is marginally more precise.

Figure 3 presents the two failure modes at their extremes on the clean subset. Panel (a) is 011_F, a translation forgery and the lowest-precision clean image on which the proposed method marked anything at all ($P = 0.1030$, $R = 0.7937$): the ground-truth region is small, 320 pixels, and the fused map marks 2467 pixels, so keypoint evidence spreads well beyond the duplicated region. Panel (b) is 194_F, a combination forgery and the lowest-recall case among clean images with a substantial ground truth (34896 pixels; $R = 0.0027$ at $P = 1.0000$): a transformation was verified, but the 94 marked pixels that survive post-processing cover a fraction of one corner of the region. These two cases bound the behaviour of the method. It is precise and incomplete, and its errors are dominated by under-marking rather than by false alarms, except where the duplicated region is very small.

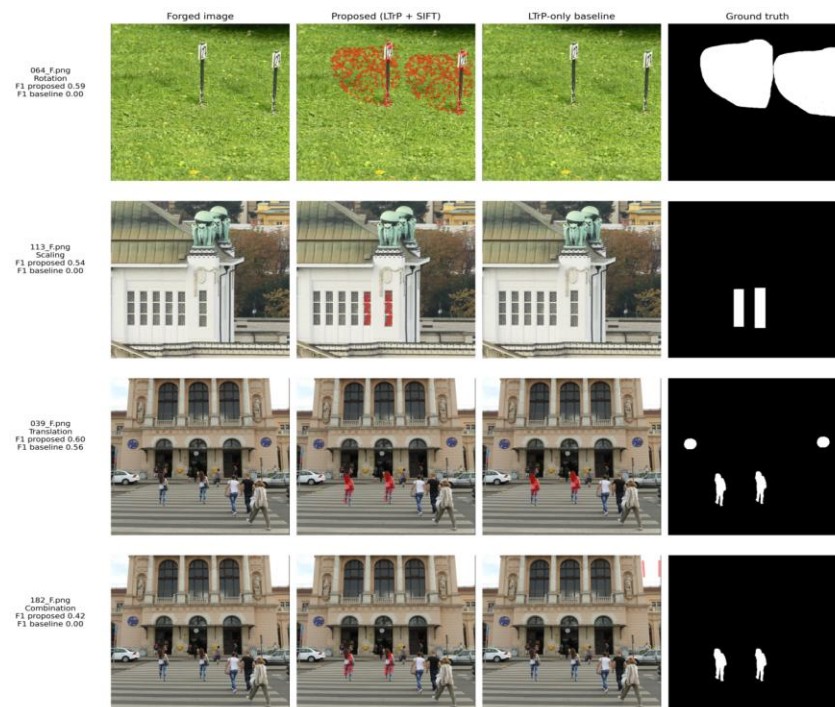


Figure 2: Localisation Maps on four Clean CoMoFoD Forgeries: 064_F (rotation), 113_F (Scaling), 039_F (Translation) and 182_F (Combination). Columns, left to Right: the Forged Image, the Map Produced by the Proposed Method, the map Produced by the LTrP-only Baseline, and the Ground-truth Mask. Row Labels give the CoMoFoD Identifier, the Manipulation Category and the Per-image F1 of both Configurations

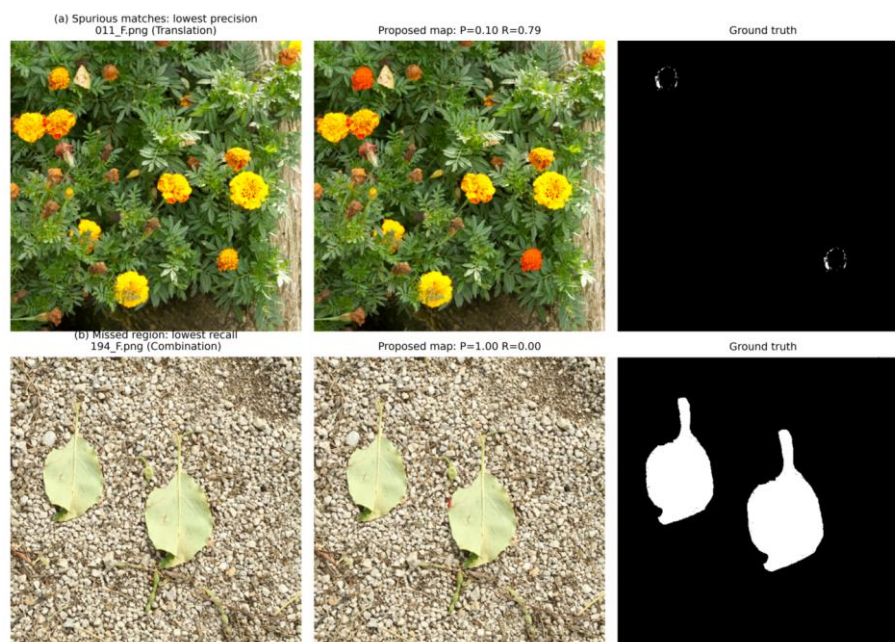


Figure 3: Failure Cases of the Proposed Method on the Clean Subset. (a) 011_F (Translation), the Lowest-precision Detected Case, $P = 0.1030$ and $R = 0.7937$, in which Keypoint Marks Spread Beyond a Small Duplicated Region. (b) 194_F (Combination), the Lowest-recall case among Images with Substantial ground Truth, $P = 1.0000$ and $R = 0.0027$. Columns, left to Right: the Forged Image, the Proposed map and the Ground Truth

Component-Wise Ablation

The ablation isolates the effect of each pipeline component. Five configurations are compared on the identical evaluation set with identical parameters: (A) LTrP blocks in RGB, with no keypoint evidence, no shift-vector filter and no post-processing beyond connected-component removal; (B) LTrP

blocks in YUV; (C) B plus shift-vector filtering and morphological post-processing, which is our re-implementation of the block-only configuration of Ganguly et al. (2023); (D) C plus symmetric SIFT gating and keypoint-block fusion, which is the proposed method; and (E) SIFT keypoints with the ratio test and RANSAC verification alone,

marked as discs of radius 10 pixels, that is, no dense block stage. Configuration E quantifies the claim that a dense block stage is necessary, while A through C quantify how much of the behaviour of the pipeline is attributable to the descriptor and its filtering rather than to the hybrid architecture. All five

were run over the same 5000 forged images and 200 originals; Table 6 reports the balanced clean subset, and the full-set F1 values are A 0.1169, B 0.1200, C 0.0947, D 0.1638 and E 0.3904.

Table 6: Component-wise Ablation on the Balanced Clean Subset (200 Forged Images and 200 Originals): mean Pixel-level F1 with Standard Deviation over Images, and Image-level Average Precision. Best Value in the F1 and AP Columns in Bold. Full-set F1 Values are Reported in the Accompanying Ablation Results

Config.	Description	Pixel-level F1	F1 std. dev.	Image-level AP
A	LTrP blocks, RGB, no keypoints, no filtering	0.2073	0.2936	0.6471
B	A with YUV conversion	0.2177	0.3434	0.7966
C	B plus shift-vector filter and morphology (LTrP-only baseline)	0.1841	0.3417	0.8705
D	C plus symmetric SIFT gating and keypoint-block fusion (proposed method)	0.2643	0.3074	0.8879
E	SIFT keypoints only, no dense block stage	0.4490	0.3354	0.8338

Configuration D, the full proposed method, achieves the highest F1 among the block-based configurations and the highest AP of the five, but not the highest F1 overall: configuration E, which uses only SIFT keypoints, reaches 0.4490 on the clean subset and 0.3904 on the full set. The claim that a dense block stage is necessary for accurate localisation, which an earlier version of this manuscript made on the strength of a placeholder value of 0.6421 for E, is not supported by our measurements; on CoMoFoD as released, the block stage as parameterised here is a net liability outside the translation category. The remaining comparisons are small relative to the per-image variability and are offered as directional only.

The YUV conversion (B against A) changes mean F1 by +0.0104 on the clean subset and +0.0031 on the full set, against standard deviations of about 0.29 and 0.20. It raises precision markedly, from 0.2719 to 0.4308, lowers recall from 0.2488 to 0.2161, and cuts the false-alarm rate on originals from 95.5% to 28.0%, which is the clearest effect attributable to it. The shift-vector filter and morphological post-processing (C against B) lower mean F1 on the clean subset, from 0.2177 to 0.1841, while raising AP from 0.7966 to 0.8705 and lowering the false-alarm rate from 28.0% to 21.0%; the two cannot be separated, since both differ between B and C, but the pattern is consistent with their intended role of trading recall for precision. Adding symmetric gating and fusion (D against C) raises mean F1 by 0.0802 and precision by 0.3836, and raises the detection rate on forged images from 41.5% to 82.5% at the cost of raising the false-alarm rate on originals from 21.0% to 30.0%. That trade is the substance of the contribution: it is favourable for detection and unfavourable for pixel-level completeness.

Configuration A, with no filtering at all, marks something on 97.0% of forged images and on 95.5% of originals, and its AP of 0.6471 is the lowest of the five. This is the expected behaviour of an unconstrained exact-match block detector on natural images: it marks self-similar texture everywhere. The chain A to D is therefore best read as a sequence of precision improvements, from 0.2719 to 0.6585 on the clean subset, purchased with recall that never exceeds 0.25. As for the descriptor itself, the strong showing of configuration E and the collapse of C under JPEG compression and additive noise mean that this study provides no evidence that LTrP is a better dense descriptor than the DCT or Zernike coefficients used in prior hybrid pipelines. Establishing that would require implementing those descriptors in the same pipeline under the same protocol, which we did not do.

The study has several limitations, and the first concerns the size of the effect it reports. The evaluation uses a single benchmark, CoMoFoD small, so generalisation to other datasets such as GRIP, MICC-F2000 or COVERAGE remains to be shown; the LTrP-only baseline is reported to be considerably stronger on GRIP than on CoMoFoD (0.9834 against 0.9093, Ganguly et al., 2023), and a single-benchmark result should not be read as evidence of general robustness. The clean subset contains only 200 images and the per-image standard deviations of mean F1 lie between 0.20 and 0.34, so several of the differences observed in the component-wise ablation are smaller than the variability of the measure and are treated above as directional rather than established.

The parameterisation of the block stage is the dominant technical limitation. Retaining LTrP coefficients at native integer magnitude and setting $\eta_{\text{simi}} = 0.002$ makes block matching an exact-match test, which fails whenever pixel values are perturbed; the category-wise results show that this causes the baseline to fail across the JPEG and noise subsets. A quantised or normalised descriptor, or a distance-based similarity criterion, could well change the ordering of configurations C, D and E, so the conclusion that the dense stage is a net liability applies only to the exact-match configuration reported here. We did not test that variant, because choosing it after seeing these results would have been a search over the evaluation set. Three design decisions that we did change during development, namely symmetric gating, keypoint-block fusion and the 5×5 closing element, were made after inspecting outputs on CoMoFoD images; the configuration was frozen once they were made, but they are not blind to the evaluation data.

The comparison is against our own re-implementations only. We could not reproduce the published LTrP-only figure from the description available, and we have removed the published values for the classical methods that appeared in earlier versions of Tables 3 and 4 rather than present numbers obtained under unknown protocols beside our own. No deep-learning detector is included, so the method is positioned against historical rather than current performance. Source and target regions are not disambiguated. The image-level metric declares a detection whenever the final map is non-empty, which is a lenient rule, and the false-alarm rates in Table 4, 30.0% of originals for the proposed method, must be read together with the detection rates rather than separately. Peak memory was not instrumented, and the reported runtime is a multi-process wall-clock figure covering all five configurations rather than a single-threaded timing for the proposed method alone.

CONCLUSION

This study evaluated a hybrid copy-move forgery detection and localisation technique combining Local Tetra Pattern (LTrP) block features with Scale-Invariant Feature Transform (SIFT) keypoint evidence using the CoMoFoD small database comprising 5,000 forged images and 200 originals. The hybrid method achieved mean pixel-level F1 scores of 0.1638 on the full dataset and 0.2643 on the clean subset, compared with 0.0947 and 0.1841, respectively, for the re-implemented LTrP-only baseline. Image-level average precision was 0.8879 on the balanced clean subset. The improvement provided by the hybrid approach was concentrated in rotation, scaling, distortion and combined transformations, as well as JPEG compression and additive noise, where SIFT keypoint evidence compensated for limitations of the exact-match LTrP block criterion. However, the hybrid performed below the LTrP-only baseline for translation, while the SIFT-only configuration outperformed both approaches on the aggregate measures. These findings demonstrate that LTrP block features and SIFT keypoints provide complementary evidence under geometrically transformed and post-processed conditions, but also show that the exact-match LTrP similarity criterion is a major constraint on overall performance. Consequently, combining block-based and keypoint-based evidence should preferably be adaptive, allowing the dominant evidence to be selected according to image characteristics. Quantisation, normalisation or distance-based matching of LTrP descriptors may improve robustness to post-processing. Future work should therefore evaluate these alternatives and validate the framework on additional benchmarks, including GRIP, MICC-F2000 and COVERAGE. Direct comparison of LTrP with DCT and Zernike descriptors under an identical pipeline would further clarify its relative contribution. Extensions to source-target disambiguation, learned feature extraction and adaptive evidence fusion may also improve the reliability and practical applicability of the proposed framework.

CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

FUNDING

This work was supported by Isa Kaita College of Education Katsina, through the TETFund intervention program on Institution Based Research (IBR).

REFERENCES

- Amerini, I., Ballan, L., Caldelli, R., Del Bimbo, A., & Serra, G. (2011). A SIFT-based forensic method for copy-move attack detection and transformation recovery. *IEEE Transactions on Information Forensics and Security*, 6(3), 1099–1110. <https://doi.org/10.1109/TIFS.2011.2129512>
- Ardizzone, E., Bruno, A., & Mazzola, G. (2015). Copy-move forgery detection by matching triangles of keypoints. *IEEE Transactions on Information Forensics and Security*, 10(10), 2084–2094.
- Bi, X., Pun, C.-M., & Yuan, X.-C. (2016). Multi-level dense descriptor and hierarchical feature matching for copy-move forgery detection. *Information Sciences*, 345, 226–242.
- Buschi, D., Berretti, S., Grana, C. M., & Tondi, B. (2021). Copy move source-target disambiguation through multi-branch CNNs. *IEEE Transactions on Information Forensics and Security*, 16, 1825–1840. <https://doi.org/10.1109/TIFS.2020.3024776>
- Chen, B., Tan, W., Coatrieux, G., Zheng, Y., & Shi, Y. Q. (2021). A serial image copy-move forgery localization scheme with source/target distinguishment. *IEEE Transactions on Multimedia*, 23, 3506–3517. <https://doi.org/10.1109/TMM.2020.3026868>
- Chen, L., Lu, W., Ni, J., Sun, W., & Huang, J. (2013). Region duplication detection based on Harris corner points and step sector statistics. *Journal of Visual Communication and Image Representation*, 24(3), 244–254.
- Christous, J., Liatsis, A., & Kotropoulos, S. (2006). Detecting copy-move forgery in digital images using localized random regions. In *Proceedings of the 17th ACM International Conference on Multimedia and Ubiquitous Computing* (pp. 195–200). <https://doi.org/10.1145/1180639.1180665>
- Cozzolino, D., Poggi, G., & Verdoliva, L. (2015). Efficient dense-field copy-move forgery detection. *IEEE Transactions on Information Forensics and Security*, 10(11), 2284–2297.
- Elaskily, M. A., Alkinani, M. H., Sedik, A., & Dessouky, M. M. (2021). Deep learning based algorithm (ConvLSTM) for copy move forgery detection. *Journal of Intelligent & Fuzzy Systems*, 40(3), 4385–4405.
- Emmanuel, S., Christodoulou, A. A., & Karatsiolis, E. (2010). KMS: A keypoint-based copy-move detection algorithm. *IEEE Transactions on Information Forensics and Security*, 5(4), 891–899. <https://doi.org/10.1109/TIFS.2010.2047653>
- Fatima, B., Ghafoor, A., Ali, S. S., & Riaz, M. M. (2022). FAST, BRIEF and SIFT based image copy-move forgery detection technique. *Multimedia Tools and Applications*, 81(30), 43805–43819.
- Fridrich, J., Soukal, D., & Lukáš, J. (2003). Detection of copy-move forgery in digital images. *Proceedings of the Digital Forensic Research Workshop*, 55–61.
- Ganguly, S., Mandal, S., Malakar, S., & Sarkar, R. (2023). Copy-move forgery detection using local tetra pattern based texture descriptor. *Multimedia Tools and Applications*, 82(13), 19621–19642. <https://doi.org/10.1007/s11042-022-14287-9>
- Kumar, N., & Meenpal, T. (2022). Salient keypoint-based copy-move image forgery detection. *Australian Journal of Forensic Sciences*. <https://doi.org/10.1111/afd.12389>
- Li, Y., & Zhou, J. (2019). Fast and effective image copy-move forgery detection via hierarchical feature point matching. *IEEE Transactions on Information Forensics and Security*, 14(5), 1307–1322.
- Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 2, 1150–1157.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>

- Murala, S., Maheshwari, R. P., & Balasubramanian, R. (2012). Local tetra patterns: A new feature descriptor for content-based image retrieval. *IEEE Transactions on Image Processing*, 21(5), 2874–2886. <https://doi.org/10.1109/TIP.2012.2188809>
- Pan, X., & Lyu, S. (2010). Region duplication detection using image feature matching. *IEEE Transactions on Information Forensics and Security*, 5(4), 857–867. <https://doi.org/10.1109/TIFS.2010.2078506>
- Ryu, S.-J., Kirchner, M., Lee, M.-J., & Lee, H.-K. (2013). Rotation invariant localization of duplicated image regions based on Zernike moments. *IEEE Transactions on Information Forensics and Security*, 8(8), 1355–1370.
- Tralić, D., Zupančić, I., Grgić, S., & Grgić, M. (2013). CoMoFoD - New database for copy-move forgery detection. *Proceedings ELMAR-2013*, 49–54.
- Wu, Y., Abd-Almageed, W., & Natarajan, P. (2018). BusterNet: Detecting copy-move image forgery with source/target localization. *Proceedings of the European Conference on Computer Vision (ECCV)*, 170–186.
- Yang, J., Liang, Z., Gan, Y., & Zhong, J. (2021). A novel copy-move forgery detection algorithm via two-stage filtering. *Digital Signal Processing*, 113, Article 103032.
- Zandi, M., Mahmoudi-Aznaveh, A., & Talebpour, A. (2016). Iterative copy-move forgery detection based on a new interest point detector. *IEEE Transactions on Information Forensics and Security*, 11(11), 2499–2512.



©2026 This is an Open Access article distributed under the terms of the Creative Commons Attribution 4.0 International license viewed via <https://creativecommons.org/licenses/by/4.0/> which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is cited appropriately.