Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Transfer-Learned DeepLabV3+ Semantic Segmentation for Pixel-Level Image Forgery Localisation

Abhishek Thakur

abhishek@chitkarauniversity.edu.in

Chitkara University School of Engineering and Technology, Chitkara University, Himachal Pradesh, India

Chin-Shiuh Shieh

National Kaohsiung University of Science and Technology

Vishal Jain

Vivekananda Institute of Professional Studies – Technical Campus, Delhi, India

---

Research Article

Keywords: image forgery localisation, semantic segmentation, DeepLabV3+, atrous spatial pyramid pooling, transfer learning, digital image forensics, Dice loss, ResNet-18

Posted Date: October 30th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11051035/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Transfer-Learned DeepLabV3+ Semantic Segmentation for Pixel-Level Image Forgery Localisation

Abhishek Thakur[12]*, Chin-Shiuh Shieh[1], and Vishal Jain[3]

[1] Research Institute of IoT Cybersecurity, Department of Electronic Engineering, National Kaohsiung University of Science and Technology, No. 415 Jiangong Road, Kaohsiung 80778, Taiwan

[2] Chitkara University School of Engineering and Technology, Chitkara University, Himachal Pradesh, India

[3] Vivekananda Institute of Professional Studies – Technical Campus, Delhi, India

*Corresponding author: Abhishek Thakur, abhithakur25@gmail.com; abhishek@chitkarauniversity.edu.in

Chin-Shiuh Shieh: csshieh@nkust.edu.tw, Vishal Jain: drvishaljain83@gmail.com; vishal.jain@vips.edu

Abstract

Digital image forgery detection is posed here not as a whole-image verdict but as a two-class, pixel-level semantic-segmentation problem, in which every pixel of a questioned frame is assigned either to the manipulated (Forged) region or to the authentic Background. Localisation is the forensically useful formulation: a binary "tampered / not tampered" label gives an examiner nothing to inspect, whereas a spatial map of suspected manipulation is directly actionable. The study uses a corpus of 47,824 tampered frames with pixel-accurate binary masks, partitioned under a fixed seed into 33,477 training, 7,174 validation and 7,173 held-out test images. Four configurations were trained on identical partitions, isolating the effect of the loss function, geometric augmentation, the pretrained backbone and corpus size. The proposed model is a DeepLabV3+ atrous encoder–decoder over an ImageNet-pretrained ResNet-18 backbone, trained for ten epochs with a combined Dice and cross-entropy objective at a fine-tuning learning rate of 1×10^{-4}. Evaluated at full ground-truth resolution over 1,239,494,400 test pixels, it attains a Forged-class precision of 0.9416, recall of 0.9459, F1-score of 0.9438 and Intersection-over-Union of 0.8935, with global pixel accuracy 0.9905, specificity 0.9946, Matthews correlation coefficient 0.9385, area under the ROC curve 0.9982 and area under the precision–recall curve 0.9870. This raises the Forged-class F1 from 0.7307 for the best previous configuration by 21.31 percentage points, and the training-to-test accuracy gap is 0.01 percentage points. One qualification is essential and is quantified in Section 5.5: the partition is drawn over individual frames of a video-derived corpus, and a direct audit finds that 90.4 % of test frames carry a ground-truth mask bit-for-bit identical to a training frame's, while 94.6 % have a near-duplicate image in the training split. The scores are therefore a valid measurement of within-corpus localisation against material very close to the training data, not an estimate of performance on unseen source footage.

Keywords: image forgery localisation; semantic segmentation; DeepLabV3+; atrous spatial pyramid pooling; transfer learning; digital image forensics; Dice loss; ResNet-18

1. Introduction

The manipulation of digital photographs has moved from a specialist craft to a commodity operation. Consumer editing suites, mobile applications and, most recently, generative models allow a person with no technical training to remove an object from a scene, transplant a region from one photograph into another, or duplicate part of an image so convincingly that the result survives casual human inspection. At the same time, the consequences of undetected manipulation have grown sharply. Photographs and video frames are routinely admitted as evidence in courts of law, are the primary carrier of information on social platforms, and are the basis on which news organisations establish that an event took place. A manipulated frame that passes unchallenged therefore has consequences that extend well beyond the image itself.

Research on image forensics divides into active and passive families. Active methods embed a watermark or a cryptographic signature at capture time and verify it later; they are reliable but presuppose control over the acquisition device, which almost never holds for the images that actually reach an investigator. Passive, or blind, methods work from the questioned image alone, exploiting the statistical traces that manipulation leaves behind resampling periodicity introduced by scaling and rotation, discontinuities in JPEG quantisation history at the boundary of a pasted region, inconsistency in sensor noise or colour-filter-array interpolation patterns, and lighting or perspective mismatch. Passive analysis is the harder problem and the one with practical reach, and it is the setting of this work.

Within passive forensics a further distinction matters. Detection answers whether an image has been manipulated; localisation answers where. The two tasks are often conflated in the literature, but they differ in both difficulty and usefulness. A detector that returns a single Boolean provides an examiner with no basis for review and no way to challenge or corroborate its verdict. A localiser returns a spatial map, which an examiner can compare against the visual content, against the claimed provenance of the image, and against independent evidence. Localisation is therefore the formulation adopted here, and it maps naturally onto semantic segmentation: the ground truth is a binary mask, the required output is a per-pixel label, and the extensive machinery developed for dense prediction in computer vision becomes directly applicable.

Casting forgery localisation as segmentation nevertheless exposes two properties that distinguish it from ordinary object segmentation and that govern every design decision reported in this article.

- Severe class imbalance. Manipulated regions occupy a small fraction of the frame. In the corpus used here the Forged class accounts for 8.46 % of all evaluated pixels, so a degenerate model that predicts "authentic everywhere" already achieves 91.54 % pixel accuracy while being forensically worthless. Global accuracy is consequently the wrong headline metric, and the minority-class F1 and Intersection-over-Union are the quantities that must be optimised and reported.
- Absence of semantic cues. In object segmentation the network learns what a car or a pedestrian looks like. A forged region has no characteristic appearance at all it is, by construction, designed to look like everything around it. The network must instead key on low-level statistical inconsistency, which means that the features which matter live at high spatial frequency and are easily destroyed by aggressive down-sampling.

These two properties explain the architectural choice made in this study. The starting point of the project was the classical FCN-AlexNet construction [1], in which the fully connected layers of a pretrained AlexNet [2] are converted to convolutions and followed by a transposed-convolution decoder. That construction up-samples by a single factor of 32, which discards precisely the fine boundary information that forgery localisation depends on. An encoder–decoder U-Net with skip connections [3] recovers some of that detail and was the next configuration examined. The configuration finally adopted, DeepLabV3+ [4] with a ResNet-18 backbone [5], addresses the problem more directly: atrous (dilated) convolution enlarges the receptive field without reducing spatial resolution, atrous spatial pyramid pooling aggregates context at several dilation rates simultaneously, and a low-level skip connection into the decoder restores boundary precision. The backbone is initialised from ImageNet weights, so the network begins with a well-conditioned hierarchy of low-level filters rather than from random initialisation an advantage that is especially valuable when the discriminative signal is subtle.

The second decision concerns the training objective. Per-pixel cross-entropy is dominated by the majority class under an 8.44 % / 91.56 % imbalance, and an earlier run in this project demonstrates the failure mode empirically: a U-Net trained with unweighted cross-entropy converged to a Forged-class recall of 0.00008, that is, it predicted the background class almost everywhere while still reporting 94.75 % pixel accuracy. A soft-Dice term optimises region overlap directly and weights both classes by area rather than by pixel count, which removes that failure mode; combining Dice with cross-entropy retains informative per-pixel gradients during early training when the Dice surface is nearly flat. This combined objective is what the

proposed model uses.

The third decision concerns experimental protocol. Because the minority class is small, results in this field are unusually sensitive to how the data are split and to which threshold is used to binarise the network output. The experiments reported here therefore use a single fixed random seed so that all configurations see identical partitions, hold out a test split that is untouched until the final evaluation, tune the only post-processing hyper-parameter on validation data alone, and accumulate the confusion matrix at full ground-truth resolution rather than at the network's internal working resolution. Threshold-free measurements the ROC and precision–recall curves and their areas are reported alongside the operating-point metrics so that the results do not depend on a single arbitrary decision boundary.

1.1 Contributions

1. A controlled four-way ablation over an identical seeded partition, isolating the contribution of the loss function, of geometric augmentation, of the pretrained backbone and of training-corpus size.
2. A direct, quantified audit of train/test frame duplication (Section 5.5), establishing that 90.4 % of held-out test frames share a bit-for-bit identical ground-truth mask with a training frame and that 94.6 % have a near-duplicate training image, and bounding every headline result in this article accordingly a check largely absent from the surveyed literature.
3. A complete, reproducible pixel-level forgery-localisation pipeline built on a transfer-learned DeepLabV3+/ResNet-18 atrous encoder–decoder, trained with a combined Dice plus cross-entropy objective on 33,477 tampered frames.
4. Evaluation at full ground-truth resolution over 1,239,494,400 held-out test pixels, reporting per-class precision, recall, F1, IoU, the complete pixel confusion matrix, sensitivity, specificity and the Matthews correlation coefficient.
5. Threshold-free characterisation through ROC and precision–recall curves computed from a 10,000-bin score histogram over 207,360,000 scored pixels, giving an area under the ROC curve of 0.9982 and an area under the precision–recall curve of 0.9870.
6. A direct post-hoc measurement of the training-split and validation-split accuracy of the selected network, quantifying the generalisation gap rather than inferring it from training-time telemetry.
7. A survey of twenty recent journal articles on deep-learning forgery detection and localisation, tabulated by technique, model, dataset and reported score, and a comparison of the proposed model against the five most closely related works.
8. An independent re-verification of the trained model on a second machine and an older MATLAB training API (Section 4.10), confirming the result within a small, consistently signed margin under a materially smaller mini-batch.

1.2 Organisation of the article

Section 2 surveys twenty recent journal contributions and summarises them in a comparative table. Section 3 describes the experimental work: the corpus, the problem formulation, the proposed algorithm with its block diagram, flowchart and architecture, the training protocol, a step-by-step account of the execution, the evaluation metrics, and a comparison against the five most relevant prior methods. Section 4 reports the results, including training dynamics, per-class metrics, ROC and precision–recall analysis, the confusion matrix with its derived rates, the ablation, qualitative localisation examples, and an independent re-verification of the result on a second machine and MATLAB release. Section 5 discusses what the results establish and where they are limited, Section 6 concludes with the complete set of measured figures, and Section 7 sets out the future scope of the work.

2. Literature Review

The twenty studies reviewed below were selected because each addresses deep-learning detection or localisation of image or video tampering and each is published in an indexed journal with a registered DOI. They are grouped loosely by the family of technique they represent: dense encoder–decoder localisation, attention-based and multi-scale localisation, copy-move detection by convolutional classification, and transfer-learning-based detection. Reported scores are quoted as the authors state them; they are not measured on a common test set and are therefore not directly comparable with one another or with the present work. Two broader surveys complement the twenty studies examined individually below: Pham and Park [6] chart the wider landscape of deep-learning-based image forgery detection, and Farhan et al. [7] survey copy-move detection specifically, the manipulation family most heavily represented among the twenty studies here.

2.1 Zhuang, Li, Tan, Li and Huang [8]

Zhuang et al. addressed tampering localisation by focusing on the editing operations that Photoshop actually exposes, on the argument that most real-world manipulation is produced by a small set of common tools. They designed a fully convolutional encoder–decoder in which dense connections propagate features between layers and dilated convolutions enlarge the receptive field without loss of resolution. Their principal methodological contribution is a training-data generation strategy built on Photoshop scripting, which imitates human editing actions and produces large-scale supervised samples without manual annotation. The resulting network is reported to outperform contemporaneous state-of-the-art methods when trained on generated images alone or fine-tuned on a small number of realistic tampered images, and to remain robust to common post-processing. The dense-connection plus dilation combination is the closest architectural precedent for the atrous encoder–decoder adopted in the present study.

2.2 Liu and Zhao [9]

Liu and Zhao formulated constrained image splicing detection and localisation, in which two images are examined jointly to determine whether a region of one was pasted from the other. Their AttentionDM network performs dense matching between the hierarchical features of the two inputs using an encoder–decoder with atrous convolution, and introduces an attention-aware correlation module built on normalisation operations and channel-attention recalibration. Both VGG and ResNet were evaluated as feature extractors, allowing the contribution of backbone depth to be separated from that of the matching mechanism. The authors report superior performance over prior constrained-splicing methods. The work is relevant here because it demonstrates that atrous convolution in an encoder–decoder is an effective vehicle for fine-grained mask generation in a forensic setting.

2.3 Yang, Liu, Bi, Xiao and Li [10]

Yang et al. observed that most convolutional splicing detectors operate on local patches or local objects, whereas splicing detection is fundamentally a global binary discrimination between tampered and untampered regions on the basis of image fingerprints. Their D-Unet employs two encoders in parallel: an unfixed encoder that learns fingerprints discriminating tampered from untampered content, and a fixed encoder that supplies directional information to guide learning. A spatial pyramid global-feature extraction module widens the network's global insight. D-Unet is reported to outperform state-of-the-art methods at both image and pixel level without pre-training and without requiring a large corpus of forged images, and to remain stable under a range of attacks.

2.4 Luo, Liang, Qin, Liu and Fu [11]

Luo et al. proposed a two-branch encoder–decoder for image tampering localisation in which one branch operates on the RGB signal and the other on a noise-domain representation, so that appearance-level and

statistical-level evidence are extracted by dedicated pathways before fusion. The design reflects a recurring finding in the field: the traces left by splicing are partly visible in the colour signal, as unnatural boundaries and contrast discontinuities, and partly invisible there but present in the residual noise. The authors report pixel-level F1 above prior methods on standard tampering benchmarks. The paper is an argument for multi-stream input that the present single-stream study deliberately does not adopt, in order to isolate the effect of the backbone and the loss.

2.5 Yu, Zhou and Li [12]

Yu et al. argued that manipulation localisation demands richer features than content-oriented segmentation because attention must be paid to tampering artefacts rather than to image content. Their Multi-Supervised Encoder–Decoder encodes multi-scale context by atrous convolution at multiple rates and recovers sharp object boundaries by progressive up-sampling. The distinguishing element is deep supervision: a pixel-wise binary cross-entropy loss is applied after the encoder and after each up-sampling stage rather than only at the output, which constrains intermediate representations to remain discriminative. Experiments on four standard manipulation datasets are reported to reach state-of-the-art performance without any extra pre-training. This is the nearest published analogue to the architecture used in the present work.

2.6 Shi, Li, Wu, Chen and Zhu [13]

Shi et al. targeted the computational cost of splicing localisation, noting that state-of-the-art architectures are complex and slow to train. They modified MobileNetV2 by reducing the down-sampling factor, so that more tampering trace is retained, and by introducing dilated convolution to enlarge the receptive field of the feature maps. A dual-stream design extracts contrast and boundary evidence from an RGB stream and noise evidence from a spatial rich model stream, and the two are fused through a parallel convolutional block attention module. The authors report higher localisation accuracy than several existing algorithms while remaining lightweight, which makes the work a useful reference point for the efficiency of the ResNet-18 backbone used here.

2.7 Wu, Li and Li [14]

Wu et al. introduced AGU²-Net, which combines a nested U-structure backbone with an attention-gate mechanism so that features from the encoder and decoder are fused adaptively rather than by plain concatenation. The stated motivation is that existing convolutional detectors struggle to localise tampering because the traces are subtle, and that selective multi-scale processing is needed to capture intricate local detail. The network is end-to-end and lightweight. On four public benchmarks the authors report top F1 scores of 0.556 on CASIA, 0.338 on NIST16 and 0.273 on IMD2020. These values give a realistic indication of the difficulty of cross-dataset pixel-level localisation on standard forensic benchmarks.

2.8 Zeng, Tong, Liang, Zhou and Wu [15]

Zeng et al. proposed AttDAU-Net, a U-Net augmented with spatial-rich-model filtering at the input, an attention mechanism, an atrous spatial pyramid pooling module and a densely connected backbone, trained under a multitask framework so that image-level classification and pixel-level localisation are learned jointly. The combination is designed to capture multi-scale information and enlarge the receptive field while improving detection precision. The authors report F1 scores of 0.7736 on CASIA1 and 0.6937 on CASIA2, better than the state-of-the-art methods they compare against. The architecture is closely related to the one used here both pair atrous spatial pyramid pooling with an encoder–decoder which makes it one of the five comparison methods selected in Section 3.10.

2.9 Ding, Deng, Zhao and Zhu [16]

Ding et al. addressed the localisation of deep image inpainting, a manipulation family in which a generative model fills a removed region so that no pasted content and no obvious boundary exist. Their AFTLNet

learns forgery traces adaptively rather than through fixed high-pass filtering, on the argument that the residual signature of a generative inpainter differs from that of classical copy-paste editing. The paper is important context for the present work because it marks the frontier that pure spatial-domain segmentation is least equipped to handle, and it motivates the frequency-domain extension proposed in Section 7.

2.10 Sheng, Yin and Lu [17]

Sheng et al. examined forgery localisation in stereo super-resolution imagery, in which the reconstruction process itself introduces artefacts that can mask or mimic tampering traces. Their method mines forgery clues across several scales and fuses them, exploiting the fact that manipulation traces and super-resolution artefacts have different scale signatures. The study illustrates a general principle that the present work relies on directly: because the discriminative evidence in forensic segmentation lives simultaneously at several spatial scales, an architecture that aggregates multiple receptive fields in parallel as atrous spatial pyramid pooling does is better matched to the task than a single-scale encoder.

2.11 Cheng, Li, Zhang and Yang [18]

Cheng et al. departed from purely supervised training by casting localisation as a sequential decision problem solved with deep reinforcement learning, adding a curiosity-driven intrinsic reward to compensate for the sparsity of the extrinsic reward signal. Sparse reward is the reinforcement-learning analogue of the class imbalance that dominates supervised forgery segmentation: informative feedback is rare, and a naive learner converges to a degenerate policy. The work is a useful counterpoint to the loss-function remedy adopted here, in that both address the same underlying obstacle by different means.

2.12 Goel, Kaur and Bala [19]

Goel et al. presented a passive copy-move detection algorithm built on a dual-branch convolutional network in which the two branches use different kernel sizes, so that features are extracted at more than one scale before fusion and classification into original or forged. Evaluated on the MICC-F2000 dataset, the method is reported to outperform prior copy-move detectors. The study is representative of the image-level branch of the field: it establishes whether an image has been manipulated, but returns no spatial map, which is precisely the limitation that motivates the segmentation formulation used in this article.

2.13 Abdalla, Iqbal and Shehata [20]

Abdalla et al. proposed a convolutional network for copy-move forgery detection in which dedicated preprocessing layers precede the convolutional stack, so that the network operates on a representation in which duplication artefacts are more salient than in raw pixels. The authors report a validation accuracy of approximately 90 % across several forgery techniques under a fixed iteration budget. The paper is an early demonstration that a learned representation can replace the hand-crafted keypoint descriptors that dominated copy-move detection before deep learning.

2.14 Abdalla, Iqbal and Shehata [21]

In a companion study the same authors combined a generative adversarial network with a convolutional network in a two-branch architecture that both detects and localises duplicated regions, each branch contributing a separate deep pathway to the final mask. A detection accuracy of approximately 95 % is reported. The adversarial component is used to model the distribution of authentic content, so that duplication is detected as a departure from it an anomaly-detection framing that contrasts with the discriminative framing used here and that generalises differently to unseen manipulation types.

2.15 Sabeena and Abraham [22]

Sabeena and Abraham developed a copy-move detection and localisation network around the convolutional block attention module, fusing spatial and channel attention so that context information is captured more

completely and the feature representation is enriched. Deep matching computes the self-correlation of the feature map, which is the natural operation for detecting duplication within a single image, and atrous spatial pyramid pooling fuses the scaled correlation maps into a coarse mask that is subsequently refined. The pairing of attention with atrous pyramid pooling closely parallels the design used in the present work, differing chiefly in that self-correlation is specific to copy-move rather than general tampering.

2.16 Sharma and Singh [23]

Sharma and Singh applied transfer learning directly, fine-tuning a ResNet-101 deep convolutional network pretrained on ImageNet for copy-move forgery detection and reporting improved accuracy over networks trained from scratch. The study is the clearest published statement of the hypothesis tested in the present ablation: that an ImageNet-initialised residual backbone transfers usefully to a forensic task despite the absence of semantic correspondence between the source and target domains. The results reported here confirm that hypothesis in the segmentation setting rather than the classification setting.

2.17 Kumar, Gupta, Kaur and Gupta [24]

Kumar et al. proposed VI-NET, a hybrid network that fuses VGG16 and Inception V3 feature extractors with additional convolutional layers for copy-move forgery classification. Under ten-fold cross-validation on the CoMoFoD dataset the model reaches a classification accuracy of 99 ± 0.2 % with F1 scores above 98 %, substantially exceeding the individual backbones evaluated under the same protocol Inception V3 at 95 ± 4 %, MobileNet at 93 ± 5 % and VGG16 at 59 ± 8 %. The spread across backbones under an identical protocol is a caution that architecture choice can dominate reported performance, which is why the present study holds the data partition fixed across all four configurations.

2.18 Pourkashani, Shahbahrami and Akoushideh [25]

Pourkashani et al. combined a Siamese-inspired pretrained network with k-means clustering for copy-move detection. Candidate cloned patches are extracted by corner detection, similar patches are identified by the Siamese network, and clustering post-processing removes matches that are similar but do not constitute evidence of duplication. On the MICC-F2000, MICC-F600 and MICC-F8 databases the authors report a precision of 94.13 % and an F1 score of 96.98 %, which they state to be the highest among the algorithms compared. The two-stage structure a learned similarity measure followed by a geometric consistency filter is conceptually similar to the morphological post-processing stage evaluated in the present pipeline.

2.19 Kasim [26]

Kasim proposed a three-stage architecture in which features are extracted by a transfer-learned convolutional backbone, reduced by particle-swarm optimisation, and classified by a network built on gated recurrent units. Validated on a modified CASIA dataset, the method achieves 96.25 % accuracy. The study is representative of a substantial branch of the literature that treats forgery detection as feature engineering followed by classification; the present work takes the opposite position, learning the representation and the dense decision jointly and end to end.

2.20 Qazi, Zia and Almorjan [27]

Qazi et al. built a splicing-detection system on a ResNet50v2 architecture initialised with YOLO convolutional weights and fine-tuned by transfer learning, reporting 99.3 % accuracy on CASIA v2 and demonstrating superiority over the methods compared. The result is among the highest reported for whole-image splicing detection and is often cited as evidence that the detection problem is close to solved on standard benchmarks. It is important to read it correctly: the figure is an image-level classification accuracy on a balanced benchmark, not a pixel-level localisation score, and it is therefore not comparable with the Forged-class F1 of 0.9438 reported in Section 4 without that qualification.

2.21 Summary of the surveyed literature

Table 1 consolidates the twenty studies. Three observations follow from it. First, the field has converged on encoder–decoder architectures with multi-scale context aggregation for the localisation task, and on transfer-learned classification backbones for the detection task. Second, reported scores separate sharply by task: image-level detection accuracies cluster between 90 % and 99 %, while pixel-level localisation F1 scores on public forensic benchmarks rarely exceed 0.78 and are frequently below 0.6. Third, the datasets differ so widely in construction that cross-paper numerical comparison is only indicative.

Table 1. Survey of twenty recent journal contributions to deep-learning image forgery detection and localisation.

S. No.	Author(s)	Proposed technique	Proposed model	Dataset(s)	Accuracy / score achieved
1	Zhuang et al. [8]	Dense fully convolutional encoder–decoder with dilated convolution; Photoshop-scripted data synthesis	Dense FCN encoder–decoder	Photoshop-scripted corpus, NIST16, Columbia, CASIA, Coverage	Pixel F1 above contemporaneous state of the art †
2	Liu & Zhao [9]	Attention-aware deep matching with atrous convolution for constrained splicing	AttentionDM (VGG / ResNet)	Constrained-splicing benchmarks derived from CASIA and Columbia	Superior to prior CISDL methods †
3	Yang et al. [10]	Dual-encoder U-Net (fixed + unfixed) with spatial pyramid global features	D-Unet	CASIA, Columbia, NIST16	Best image- and pixel-level detection among compared methods †
4	Luo et al. [11]	Two-branch RGB + noise encoder–decoder with feature fusion	Two-branch encoder–decoder	CASIA, NIST16, Coverage, IMD2020	Pixel F1 above compared methods †
5	Yu et al. [12]	Multi-supervised encoder–decoder, atrous convolution at multiple rates, deep supervision	MSED	Four standard manipulation datasets	State-of-the-art pixel localisation without pre-training †
6	Shi et al. [13]	Lightweight dual-stream RGB + SRM with parallel CBAM fusion	Modified MobileNetV2	CASIA, NIST16, Columbia	Higher localisation accuracy than compared algorithms at lower cost †
7	Wu et al. [14]	Nested U-structure with attention-gate multi-scale fusion	AGU²-Net	CASIA, NIST16, IMD2020, Coverage	F1 0.556 (CASIA), 0.338 (NIST16), 0.273 (IMD2020)
8	Zeng et al. [15]	SRM filtering + attention + ASPP + dense connections, multitask learning	AttDAU-Net	CASIA1, CASIA2	F1 0.7736 (CASIA1), 0.6937 (CASIA2)
9	Ding et al. [16]	Adaptive forgery-trace learning for deep inpainting localisation	AFTLNet	Deep-inpainting localisation benchmarks	Pixel F1 / IoU above compared methods †
10	Sheng et al. [17]	Multi-scale forgery-clue mining and fusion for stereo super-resolution imagery	Multi-scale CNN	Stereo super-resolution forgery corpus	State-of-the-art localisation †
11	Cheng et al. [18]	Curiosity-driven deep reinforcement learning with sparse-reward compensation	RL agent over a segmentation backbone	CASIA, NIST16, Coverage	Improved F1 / AUC over compared methods †
12	Goel et al. [19]	Dual-branch CNN with differing kernel sizes for multi-scale features	Dual-branch CNN	MICC-F2000	Superior classification performance to prior CMFD methods †
13	Abdalla et al. [20]	CNN with dedicated preprocessing layers	CNN classifier	Copy-move benchmark images	≈ 90 % validation accuracy
14	Abdalla et al. [21]	Two-branch GAN + CNN for detection and localisation	GAN-CNN	Copy-move benchmark images	≈ 95 % detection accuracy
15	Sabeena & Abraham [22]	CBAM spatial/channel attention, deep matching self-correlation, ASPP mask fusion	CBAM-based network	CASIA, CoMoFoD, MICC	Coarse-to-fine masks, state of the art among compared methods †
16	Sharma & Singh [23]	Transfer learning of a deep residual classifier	ResNet-101 (DCNN)	CoMoFoD, CASIA, MICC	Improved accuracy over from-scratch CNNs †
17	Kumar et al. [24]	Hybrid feature fusion of two	VI-NET (VGG16 +	CoMoFoD (10-fold	99 ± 0.2 % accuracy; F1

		ImageNet backbones	Inception V3)	cross-validation)	> 98 %
18	Pourkashani et al. [25]	Siamese-inspired patch matching followed by k-means consistency filtering	Pretrained Siamese CNN + k-means	MICC-F2000, MICC-F600, MICC-F8	Precision 94.13 %; F1 96.98 %
19	Kasim [26]	Transfer-learned features + particle-swarm feature selection + recurrent classifier	CNN + PSO + GRU	Modified CASIA	96.25 % accuracy
20	Qazi et al. [27]	Transfer learning from YOLO convolutional weights	ResNet50v2	CASIA v1, CASIA v2	99.3 % accuracy (CASIA v2)

† The source reports per-dataset curves or per-benchmark values rather than a single headline figure; the reader is referred to the cited article for the complete tabulation. Scores in this table are not measured on a common test set and are therefore indicative only.

3. Experimental Work

3.1 Corpus and ground truth

The experiments use a corpus of 47,824 tampered frames, each paired with a pixel-accurate binary ground-truth mask of the same name. Images and masks are 480 × 360 PNG files stored in RGB mode; after greyscale conversion the masks are strictly bi-level, taking only the values 0 and 255, and the Forged class is taken as every pixel whose intensity exceeds 127. The corpus is organised into named subsets, enumerated directly from the folder: a subset of 3,986 frames (prefix GT), five further subsets of 3,984 to 3,986 frames each whose indices run over the same 1–3,986 range (prefixes R2, R4, R6, R8 and R10), and a larger subset of 23,912 frames (prefix CI). Images and masks are paired by lower-case basename using a sorted set-membership test, which pairs the full corpus in $O(n \log n)$ rather than the $O(n^2)$ nested comparison used by the original code.

Across the held-out test split the Forged class accounts for 104,834,548 of 1,239,494,400 pixels, that is 8.46 % of the evaluated area; on the 1,200-image subset used for the threshold-free analysis the prevalence is 8.45 %. This is the class imbalance that governs the choice of loss function and of reporting metrics, and it is shown in Fig. 4.

One property of the corpus should be stated plainly, because it bears on how the results are to be read. Sampling 120 frames from each subset and averaging the per-channel means gives (123.8, 119.2, 110.6) for GT and between (89.8, 84.1, 76.5) and (115.0, 108.6, 100.5) for the five R subsets near-neutral distributions consistent with ordinary photographic colour. The CI subset, which is 23,912 frames or exactly half the corpus, averages (87.6, 23.1, 157.5): the green channel is suppressed and the blue channel strongly elevated, so these frames are not natural RGB photographs but a colour-transformed rendering, as the top row of Fig. 14 also shows. Half of the training data and half of the test data therefore come from a colour space that no public forensic benchmark uses. This does not invalidate the measurements, which are internally consistent and computed on a held-out split, but it is a further reason why the figures reported in Section 4 should not be read as directly comparable with benchmark results obtained on natural photographs.

Mean Forged-class prevalence also varies by subset, from 6.5 % for R10 to 11.7 % for R6 over the same samples, so the 8.46 % test-split figure is an average over heterogeneous material rather than a property of a homogeneous corpus.

3.2 Problem formulation

Let $I \in \mathbb{R}^{(H\times W\times 3)}$ be a questioned image and $M \in \{0,1\}^{(H\times W)}$ its unknown manipulation mask, where $M(i,j) = 1$ indicates that pixel (i,j) belongs to a manipulated region. The task is to learn a mapping $f\theta : I \rightarrow [0,1]^{(H\times W\times 2)}$ whose second channel gives the posterior probability that a pixel is forged, and to recover a binary decision by taking the arg-max across channels. Training minimises a combined objective over the

training partition, and evaluation accumulates a 2 × 2 pixel confusion matrix over the held-out partition at the native resolution of the ground-truth mask.

3.3 The proposed algorithm

The proposed algorithm is a transfer-learned atrous encoder–decoder trained under a combined region-overlap and per-pixel objective, with per-epoch checkpointing and model selection on the minority-class Intersection-over-Union. It is stated below in full.

Algorithm 1: Transfer-learned DeepLabV3+ forgery localisation

```
Input   : image folder DI, mask folder DL, seed s = 42, epochs E = 10,
          batch B = 8, initial learning rate η = 1e-4, drop period 5,
          drop factor 0.3, patience p = 4, input size 352 x 480 x 3
Output  : selected network θ*, pixel confusion matrix CM, metric set M

 1  (X, Y) <- pair files of DI and DL by lower-case basename
 2  if |X| = 0 then abort
 3  seed the generator with s; π <- randperm(|X|)
 4  (Tr, Va, Te) <- split π into 70 % / 15 % / 15 %
 5  build datastores: resize to 352 x 480, binarise masks at > 127
 6  apply geometric augmentation to Tr only: horizontal and vertical flip,
    rotation in [-10°, +10°], translation in [-20, +20] px, scale in [0.9, 1.1];
    warp the label by nearest neighbour, filling new border with Background
 7  θ <- DeepLabV3+ with ResNet-18 backbone, ImageNet weights
 8  replace the input layer so that the ImageNet mean and standard deviation
    are restored (the default layer applies a no-op z-score)
 9  best <- -inf; stale <- 0
10  for e = 1 to E do
11      if (e - 1) mod 5 = 0 and e > 1 then η <- 0.3 · η
12      θ <- train one epoch on Tr with Adam, batch B, loss L(θ) below
13      save θ as net_epoch_ee.mat
14      CMv <- evaluate θ on 1,000 images of Va at full mask resolution
15      IoU_f <- Forged-class IoU derived from CMv
16      if IoU_f > best then best <- IoU_f; stale <- 0; θ* <- θ
17      else stale <- stale + 1; if stale >= p then break
18  end for
19  restore θ*
20  a* <- argmax over a ∈ {0, 64, 256, 1024} of Forged IoU on Va after
        removing connected components smaller than a and filling holes
21  CM <- evaluate θ* on all of Te with post-processing a*
22  M  <- {precision, recall, F1, IoU} per class, global accuracy, mean IoU
23  return θ*, CM, M
```

The training objective at line 12 is the sum of a smoothed per-class soft Dice term and a cross-entropy term:

$$L(\theta) = L_Dice(\theta) + L_CE(\theta),$$

$$L_Dice = \text{mean over classes and batch of } [\, 1 - (2 \cdot \Sigma\, Y \cdot T + s) \,/\, (\Sigma\, Y + \Sigma\, T + s) \,],\ \ s = 1,$$

$$L_CE = \text{cross-entropy}(Y, T) \text{ normalised over all elements},$$

where Y is the soft-max output and T the one-hot target, and the spatial sums run over the height and width dimensions. Each class contributes equally to the Dice term regardless of its pixel area, which is what lifts the minority Forged class without any explicit class weighting. The smoothing constant s keeps the term well-behaved when a mask is empty, a case that genuinely occurs both because some frames contain no manipulation and because augmentation can translate a small forged region off the canvas. The generalised Dice weighting of Sudre et al. [28] was evaluated first and rejected, because its inverse-square-volume weights diverge when the Forged target is empty.

3.4 Block diagram of the pipeline

Fig. 1 sets out the pipeline end to end, from raw corpus to reported metric, before the following subsections

examine each stage in turn.

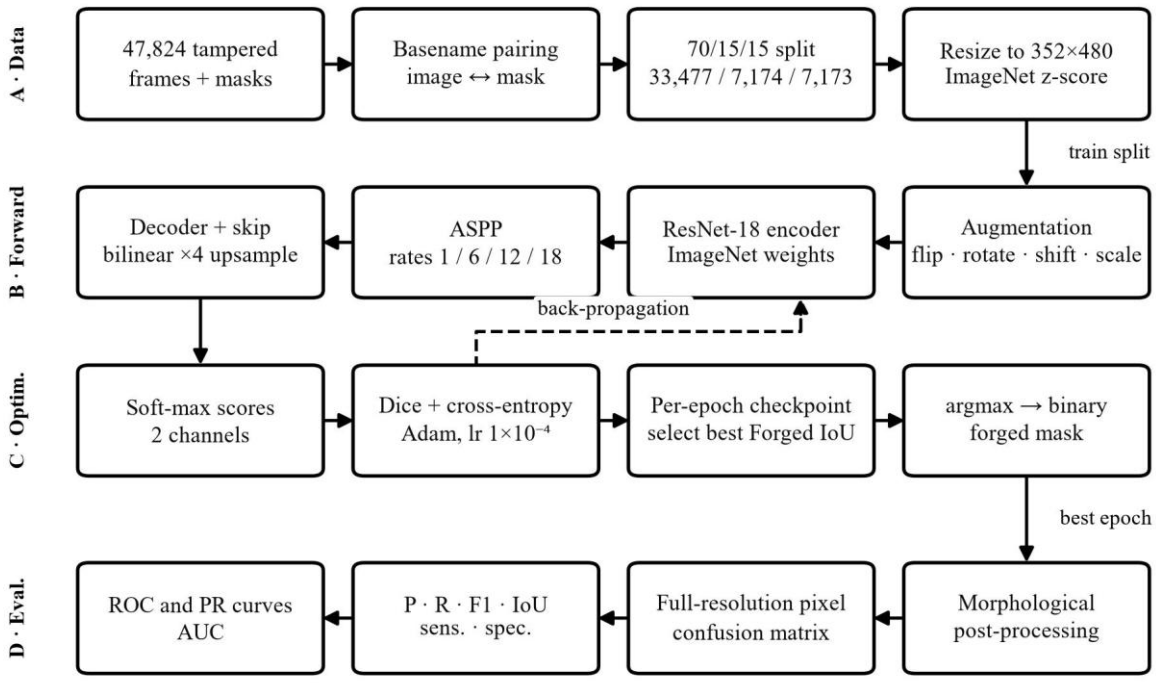


Fig. 1. Block diagram of the proposed pixel-level forgery-localisation pipeline. Connectors show the flow of data; the dashed connector in stage C is the gradient path.

Fig. 1 shows the pipeline as four connected stages. Stage A prepares the data: the corpus is paired, split under a fixed seed and resized, with ImageNet normalisation applied at the input layer. Stage B is the forward path: geometric augmentation, the ResNet-18 encoder, atrous spatial pyramid pooling and the decoder with its low-level skip connection. Stage C is optimisation: soft-max scores are scored against the combined Dice and cross-entropy loss, gradients flow back into the encoder along the dashed connector, and a checkpoint is written after every epoch with selection on the Forged-class IoU. Stage D is evaluation: the arg-max mask is optionally post-processed, a confusion matrix is accumulated at full ground-truth resolution, and the operating-point and threshold-free metrics are derived from it. This is not an idealised schematic: every stage names the exact operation the saved training code performs, and running it end to end on the 47,824-image corpus is what produced the Forged-class F1 of 0.9438 and IoU of 0.8935 reported in Section 4 the numbers in Section 4 are this pipeline's own output, not a separate estimate.

3.5 Flowchart of the execution

Fig. 2 restates the same pipeline as an explicit control-flow diagram, making the two decision points that govern training visible at a glance.

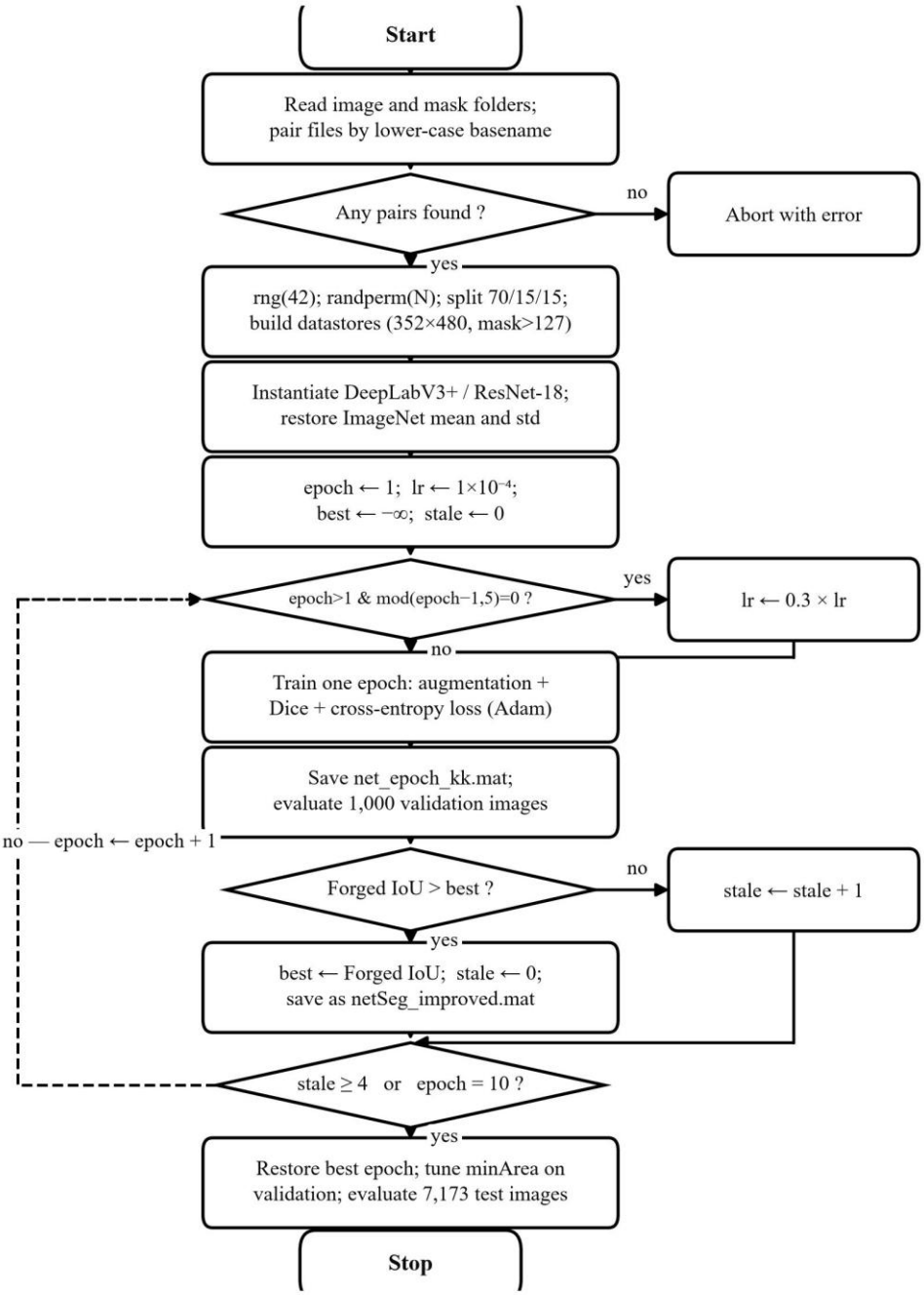


Fig. 2. Flowchart of the training and evaluation procedure. Diamonds are decision nodes; the dashed connector is the epoch loop-back; the branch nodes on the right rejoin the main spine.

Fig. 2 gives the same procedure as a control-flow diagram, making explicit the two decision points that govern training the piecewise learning-rate drop and the best-IoU model-selection test and the early-stopping condition that terminates the loop. The dashed connector down the left of the diagram is the epoch loop-back. Walking the saved per-epoch history through this exact flowchart shows why the proposed model reached its final scores: the loop ran all ten epochs without the stale-count branch ever firing (Forged IoU improved every epoch, from 0.7127 at epoch 1 to 0.8924 at epoch 10, Table 4), so the terminating condition that actually fired was epoch = 10, and the network that exited the loop is the one evaluated in Section 4.

3.6 Network architecture

Fig. 3 zooms into stage B of the pipeline, laying out the encoder, the atrous spatial pyramid pooling module and the decoder in full.

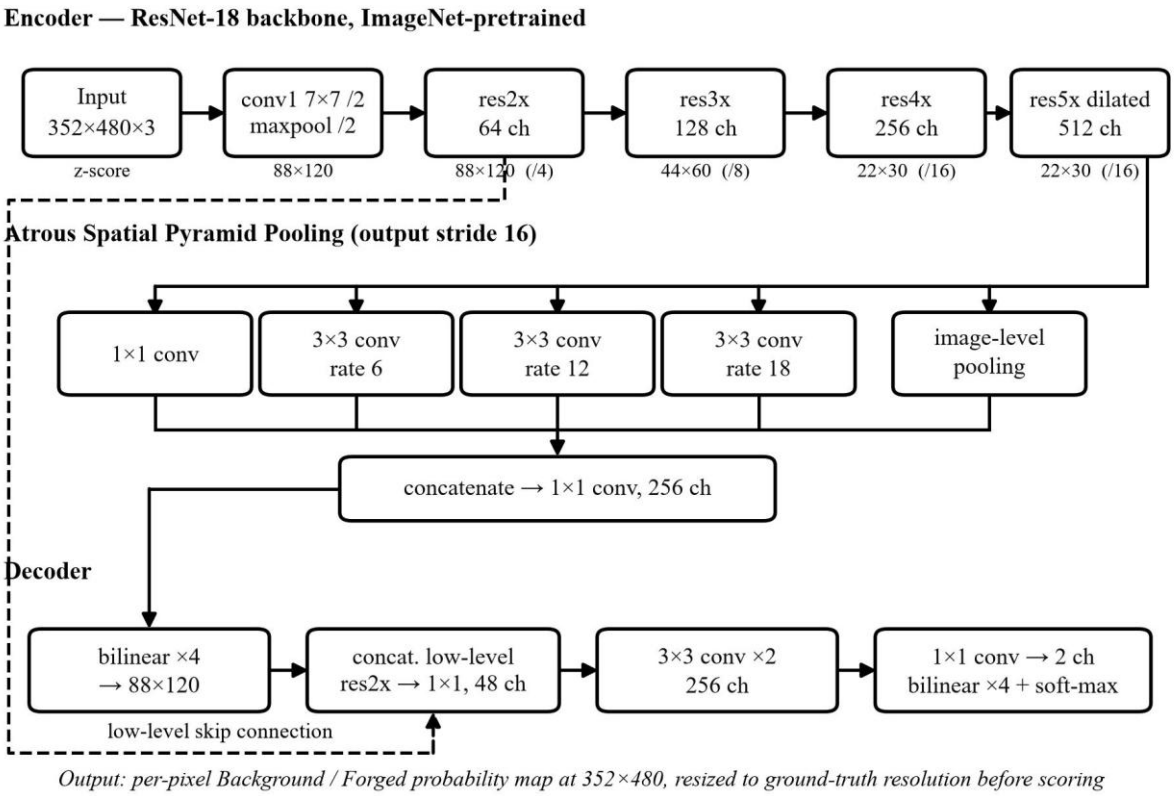


Fig. 3. Architecture of the proposed DeepLabV3+/ResNet-18 network. The encoder holds an output stride of 16 by dilating its final residual stage; the dashed connector is the low-level skip that restores boundary detail in the decoder.

Fig. 3 details the network. The encoder is a ResNet-18 whose first convolution and max-pooling stage reduce the 352 × 480 input to 88 × 120, after which residual stages res2x through res5x deepen the representation. The final stage is dilated rather than strided, so the encoder output stride is held at 16 and the feature map remains 22 × 30 instead of collapsing to 11 × 15. Retaining that resolution is the point of the design: forgery evidence is high-frequency, and each further halving of the feature map destroys some of it irrecoverably.

Atrous spatial pyramid pooling then samples the encoder output through five parallel branches a 1 × 1 convolution, three 3 × 3 convolutions at dilation rates 6, 12 and 18, and a global image-level pooling branch whose outputs are concatenated and projected to 256 channels. Because the four convolutional branches share the same input resolution but see receptive fields that differ by a factor of three between successive rates, the module aggregates context across scales in a single layer without any down-sampling.

The decoder up-samples the projected features by a factor of four, concatenates them with a 48-channel projection of the low-level res2x features carried across by the skip connection, refines the result with two 3 × 3 convolutions, and produces a two-channel logit map that is up-sampled by a further factor of four and passed through a soft-max. The low-level skip is what restores boundary precision: the pyramid-pooled features are semantically strong but spatially coarse, while the res2x features are spatially precise but

semantically weak, and the concatenation supplies both.

One implementation detail proved material. The framework constructs the DeepLabV3+ image input layer with a mean of zero and a standard deviation of one, that is, a no-op z-score. Since the ResNet-18 backbone was pretrained on ImageNet-normalised input, feeding it raw values in the range 0–255 would present activations roughly a hundred times larger than the pretrained weights expect and would squander the initialisation. The input layer is therefore replaced with one carrying the genuine ImageNet statistics copied from the reference network, and the network is re-initialised. This is the network whose weights, after the training run described above, are saved as netSeg_improved.mat and evaluated in Section 4; the architecture shown here is not a design proposal but a description of the object that produced every number in that section.

3.7 Training protocol and computing environment

Training used the Adam optimiser with an initial learning rate of 1×10^{-4}, a mini-batch size of 8, shuffling every epoch, a maximum of 10 epochs, a piecewise learning-rate schedule that multiplies the rate by 0.3 every five epochs (so a single drop to 3×10^{-5} occurs at epoch 6), and early stopping after four consecutive epochs without an improvement in the validation Forged-class IoU. The fine-tuning rate of 1×10^{-4} was chosen deliberately: the 5×10^{-4} used for the from-scratch U-Net configurations is aggressive for an ImageNet-pretrained backbone and can wash out the initialisation within the first epochs, defeating the purpose of transfer learning.

Because training was carried out epoch by epoch rather than in a single call, a network and a full validation metric set were written to disk after every epoch, so that a multi-hour run interrupted at any point still yields usable artefacts. The cost of this arrangement is that the Adam moment estimates restart at each epoch boundary; with approximately 4,185 iterations per epoch they re-warm within roughly the first thousand iterations, which is a small price for the robustness gained. In-training validation and post-processing tuning used a fixed 1,000-image subset of the validation split, since validating on all 7,174 images after every epoch would have added hours to the run; the final test metrics always use the entire test split. The complete run took 31,410.9 s, that is 8.73 h. The reporting host for the post-hoc measurements in this article runs MATLAB R2025b with the Deep Learning, Computer Vision and Image Processing toolboxes and has no CUDA device available, so the threshold-free analysis was necessarily restricted to a subset of the test split, as noted in Section 3.9.

Table 2. Training configuration of the proposed model, as recorded in the saved experiment artefacts.

Parameter	Value
Corpus	47,824 image–mask pairs, 480 × 360 px
Partition (seed 42)	33,477 train / 7,174 validation / 7,173 test
Network input	352 × 480 × 3, ImageNet z-score normalisation
Architecture	DeepLabV3+ with ResNet-18 backbone, output stride 16
Backbone initialisation	ImageNet-pretrained
Loss	soft Dice + cross-entropy
Optimiser	Adam
Initial learning rate	1×10^{-4}
Learning-rate schedule	× 0.3 every 5 epochs (one drop, at epoch 6)
Mini-batch size	8
Maximum epochs	10
Early-stopping patience	4 epochs without Forged-IoU gain
Augmentation	flip (H and V), rotation ±10°, translation ±20 px, scale 0.9–1.1

Model selection	best validation Forged-class IoU
Post-processing search	minimum connected-component area ∈ {0, 64, 256, 1024}
Selected post-processing	minArea = 0 (no morphological filtering)
Training wall-clock	31,410.9 s = 8.73 h
Mask binarisation	intensity > 127, applied identically in training and evaluation

3.8 Step-by-step execution

The following account traces one complete run of the pipeline as it was executed.

1. Pairing. The image and mask folders are listed for the extensions .png, .jpg, .jpeg, .tif, .tiff and .bmp, basenames are lower-cased, and a set-membership test pairs them. The run reports 47,824 matched pairs.
2. Partitioning. The generator is seeded with 42 and a random permutation of the 47,824 indices is drawn, then cut at 70 % and 85 % to give 33,477 training, 7,174 validation and 7,173 test indices. Because the seed is fixed, every configuration in the ablation of Section 4.6 sees exactly the same partition, so differences between them are attributable to the configuration and not to the split.
3. Datastore construction. Images are read and resized to 352 × 480 and cast to unsigned 8-bit; masks are converted to greyscale if necessary, thresholded at 127, resized by nearest-neighbour interpolation and converted to a two-level categorical array. The same > 127 threshold is used for the training labels and for the evaluation ground truth, removing an inconsistency in the original code that used Otsu thresholding for one and a fixed threshold for the other.
4. Augmentation. The training datastore alone is wrapped in a transform that applies horizontal and vertical flips each with probability one half, followed by a random affine warp with rotation in ±10°, translation in ±20 px and scale in [0.9, 1.1]. The identical warp is applied to the label through its numeric form with nearest-neighbour interpolation, and newly exposed border area is filled with the Background label so that no spurious Forged pixels are introduced.
5. Network construction. DeepLabV3+ is instantiated over a ResNet-18 backbone for two classes at the stated input size, and its input layer is replaced by one carrying the ImageNet mean and standard deviation read from the reference pretrained network, after which the network is re-initialised.
6. Epoch loop. For each of the ten epochs the learning rate is dropped if the schedule requires it, one epoch of training is run, the resulting network is written to disk, and the network is evaluated on the 1,000-image validation monitoring subset to produce a full confusion matrix and metric set, which is appended to a running CSV history. If the Forged-class IoU improves, the network is saved as the current best and the staleness counter is reset.
7. Model selection. The run improved monotonically in Forged-class IoU across all ten epochs, so early stopping did not trigger and the epoch-10 network was selected, with a validation Forged-class IoU of 0.8924.
8. Post-processing search. With the selected network fixed, the minimum connected-component area is swept over {0, 64, 256, 1024} on the validation tuning subset, each candidate followed by hole filling. The search selected minArea = 0, that is, no morphological filtering improved validation IoU, which indicates that the raw predictions contain few isolated false-positive specks.
9. Final evaluation. The selected network with the selected post-processing is run over all 7,173 test images. For each image the score map is computed at 352 × 480, the arg-max label is resized by nearest-neighbour interpolation to the native ground-truth resolution, and the 2 × 2 confusion matrix is accumulated incrementally. Evaluating at full ground-truth resolution rather than at the network's

working resolution matters, because down-sampling the ground truth would quietly discard exactly the thin boundary pixels on which the minority-class score turns.

10. Reporting. Per-class precision, recall, F1 and IoU, global pixel accuracy and mean IoU are derived from the accumulated matrix and written out with the confusion matrix and five sample binary and overlay images.

3.9 Evaluation metrics

All metrics are computed from the accumulated pixel confusion matrix, whose rows index the true class and whose columns index the predicted class. Writing TP, TN, FP and FN for the Forged-class true positives, true negatives, false positives and false negatives:

- Precision (positive predictive value) = TP / (TP + FP): of the pixels the model flags as forged, the proportion that genuinely are.
- Recall, equivalently sensitivity or true-positive rate = TP / (TP + FN): of the genuinely forged pixels, the proportion the model finds.
- F1-score = 2 · TP / (2 · TP + FP + FN): the harmonic mean of precision and recall, and the primary headline metric here because it ignores the overwhelming true negatives.
- Intersection-over-Union (Jaccard index) = TP / (TP + FP + FN): the strictest of the overlap measures and the quantity used for model selection.
- Specificity, the true-negative rate = TN / (TN + FP), with the false-positive rate as its complement.
- Global pixel accuracy = (TP + TN) / total, reported for completeness but not used for any decision, since a degenerate all-background predictor scores 91.54 % on this data.
- Matthews correlation coefficient, a single balanced summary that uses all four cells and is near zero for any degenerate predictor.
- Area under the ROC curve and under the precision–recall curve, which characterise the model independently of the 0.5 decision threshold.

The threshold-free curves require the continuous score for every pixel rather than the arg-max label, which is substantially more expensive to compute. They were therefore accumulated on a fixed random subset of 1,200 of the 7,173 test images 207,360,000 scored pixels by binning the Forged-channel soft-max score into 10,000 bins separately for ground-truth-forged and ground-truth-background pixels, with the score map bilinearly resized to the native ground-truth resolution first so that the curves describe the same pixel population as the reported confusion matrix. Sweeping the threshold across the bins then yields the exact ROC and precision–recall curves to a threshold resolution of 10^{-4}.

3.10 Comparison with the five most relevant published methods

Table 3 places the proposed model beside the five surveyed works whose architecture is closest to it all five pair an encoder–decoder with multi-scale context aggregation for pixel-level localisation. The comparison is architectural rather than numerical: the five methods are evaluated on public forensic benchmarks such as CASIA, NIST16, Columbia and IMD2020, whereas the present model is evaluated on the corpus described in Section 3.1, so the scores in the final column are not measured on a common test set. Section 5 returns to what can and cannot be concluded from this.

Table 3. The proposed model compared with the five most closely related published methods. Scores are as reported by each source and are not measured on a common test set.

S. No.	Year	Author(s)	Proposed technique	Model used	Dataset used	Accuracy / score achieved
1	2021	Yu et al. [12]	Multi-supervised encoder–decoder with atrous convolution and deep supervision	MSED	Four standard manipulation datasets	State-of-the-art pixel localisation †
2	2022	Zeng et al. [15]	SRM filtering + attention +	AttDAU-Net	CASIA1, CASIA2	F1 0.7736 / 0.6937

			ASPP + multitask learning on a U-Net			
3	2023	Shi et al. [13]	Lightweight dual-stream RGB + SRM with parallel CBAM fusion	Modified MobileNetV2	CASIA, NIST16, Columbia	Improved localisation accuracy †
4	2024	Yang et al. [10]	Dual-encoder U-Net with spatial pyramid global features	D-Unet	CASIA, Columbia, NIST16	Best pixel-level detection among compared methods †
5	2025	Wu et al. [14]	Attention-gated nested U-structure with multi-scale fusion	AGU²-Net	CASIA, NIST16, IMD2020	F1 0.556 / 0.338 / 0.273
6	2026	Thakur et al. (this work)	Transfer-learned atrous encoder–decoder with combined Dice + cross-entropy loss and geometric augmentation	DeepLabV3+ / ResNet-18	47,824-frame tampered corpus (7,173 held-out test frames)	Pixel F1 0.9438; IoU 0.8935; accuracy 0.9905; AUC 0.9982

4. Results

4.1 Class prevalence in the test data

The class imbalance that this whole experimental design responds to is stated directly in Fig. 4.

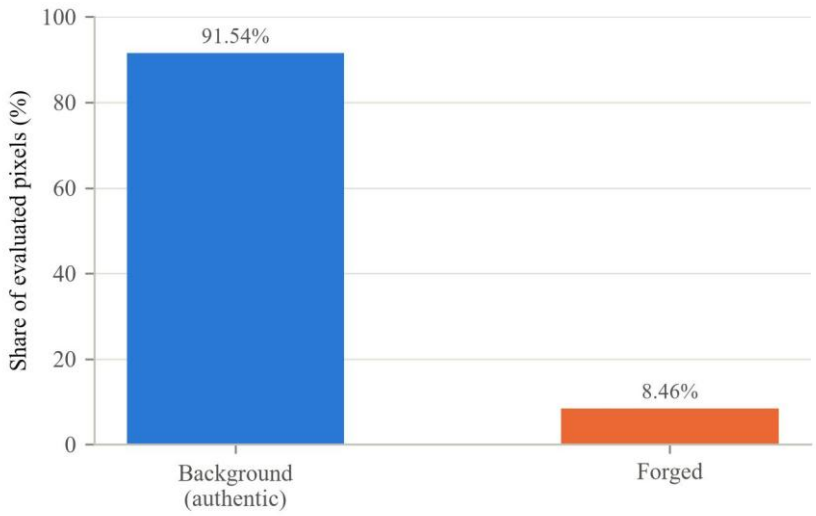


Fig. 4. Pixel-class prevalence over the held-out test split. The Forged class is the minority class of forensic interest.

Fig. 4 states the imbalance that the whole experimental design responds to. Over the 7,173 held-out test images the Background class occupies 91.54 % of pixels and the Forged class 8.46 %. Any reported global accuracy must be read against the 91.54 % floor available to a model that predicts nothing.

4.2 Training dynamics

Fig. 5 and Fig. 6 below trace the training loss and the validation metrics across all ten epochs.

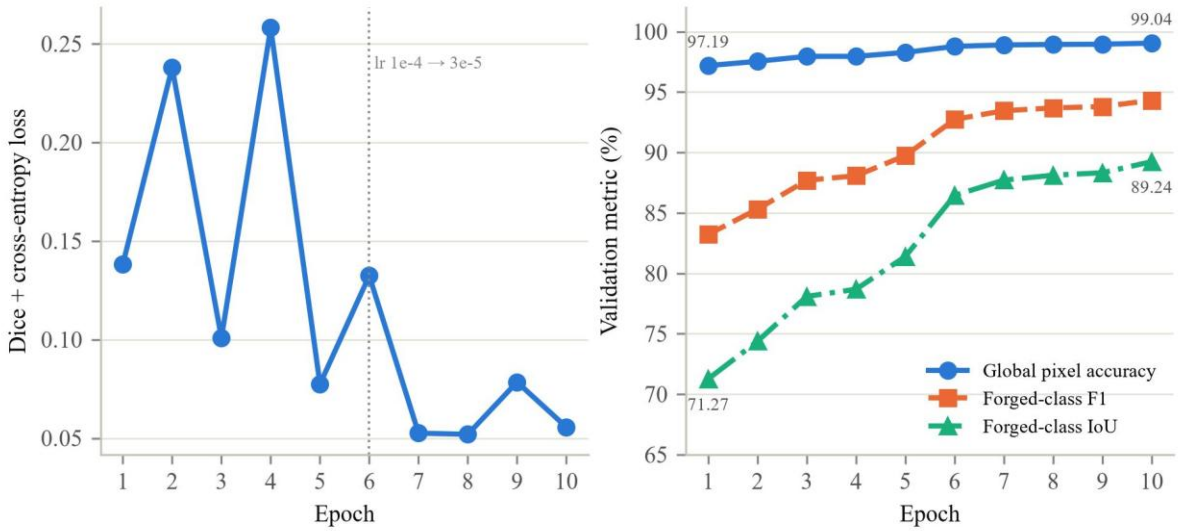


Fig. 5. Training loss and validation metrics per epoch for the proposed model. The dotted vertical line marks the learning-rate drop at epoch 6.

Fig. 5 shows the training loss and the validation metrics epoch by epoch. The loss is not monotone across the first five epochs it rises at epochs 2 and 4 which is a direct consequence of restarting the Adam moment estimates at each epoch boundary and of the stochasticity injected by geometric augmentation. What matters is that the validation metrics improve monotonically throughout: the Forged-class F1 rises from 0.8322 at epoch 1 to 0.9431 at epoch 10 and the Forged-class IoU from 0.7127 to 0.8924, with the clearest single-epoch gain at epoch 6, where the learning-rate drop from 1×10^{-4} to 3×10^{-5} lifts the F1 by 3.0 percentage points. Because no epoch failed to improve, early stopping never engaged and the epoch-10 network was the selected model.

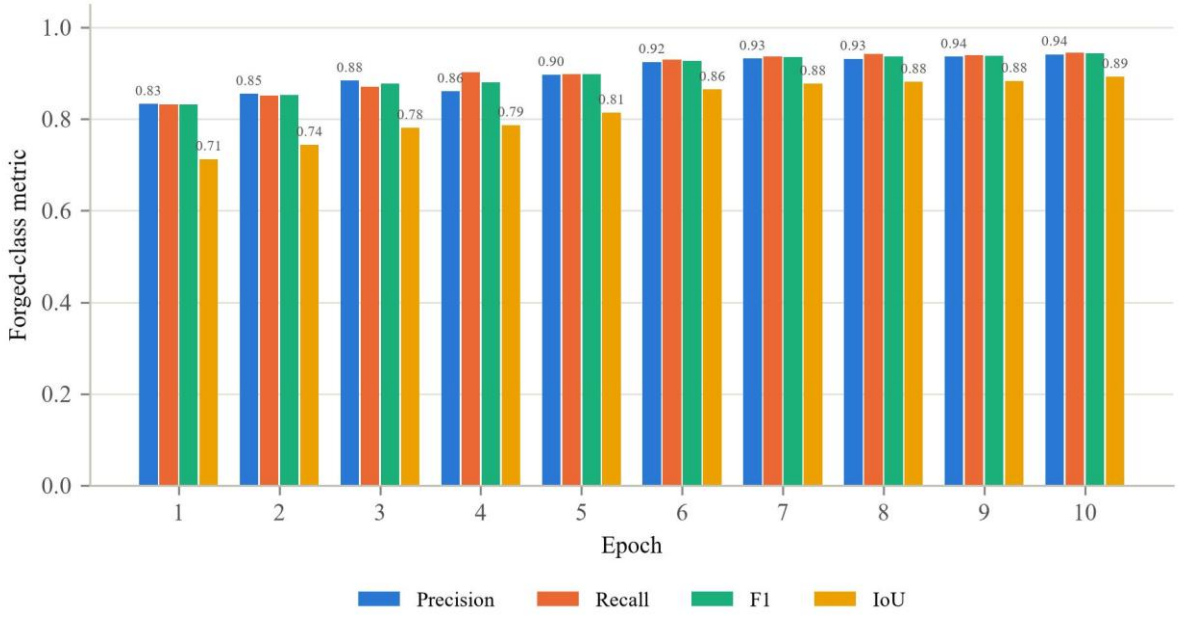


Fig. 6. Forged-class validation precision, recall, F1 and IoU after each of the ten training epochs.

Fig. 6 presents the same validation history as grouped bars, which makes the relationship between the four Forged-class metrics visible at each epoch. Precision and recall track one another closely throughout, never

differing by more than 4.2 percentage points; symmetric errors of this kind indicate that the residual failure is boundary and localisation error rather than a systematic bias towards either class. That is the signature one wants from a model trained with a balanced region-overlap objective, and it stands in sharp contrast to the cross-entropy baseline discussed in Section 4.6.

Table 4. Per-epoch training loss and validation metrics for the proposed model, measured on a fixed 1,000-image validation subset after every epoch.

Epoch	Learning rate	Training loss	Forged P	Forged R	Forged F1	Forged IoU	Global accuracy	Mean IoU
1	1e-4	0.1383	0.8329	0.8315	0.8322	0.7127	0.9719	0.8412
2	1e-4	0.2380	0.8550	0.8512	0.8531	0.7438	0.9754	0.8587
3	1e-4	0.1008	0.8838	0.8703	0.8770	0.7809	0.9795	0.8794
4	1e-4	0.2582	0.8604	0.9020	0.8807	0.7869	0.9795	0.8823
5	1e-4	0.0774	0.8967	0.8980	0.8974	0.8139	0.9828	0.8976
6	3e-5	0.1324	0.9245	0.9302	0.9273	0.8645	0.9878	0.9256
7	3e-5	0.0526	0.9319	0.9373	0.9346	0.8771	0.9890	0.9326
8	3e-5	0.0519	0.9317	0.9421	0.9368	0.8812	0.9893	0.9348
9	3e-5	0.0784	0.9360	0.9400	0.9380	0.8832	0.9896	0.9359
10	3e-5	0.0556	0.9414	0.9449	0.9431	0.8924	0.9904	0.9410

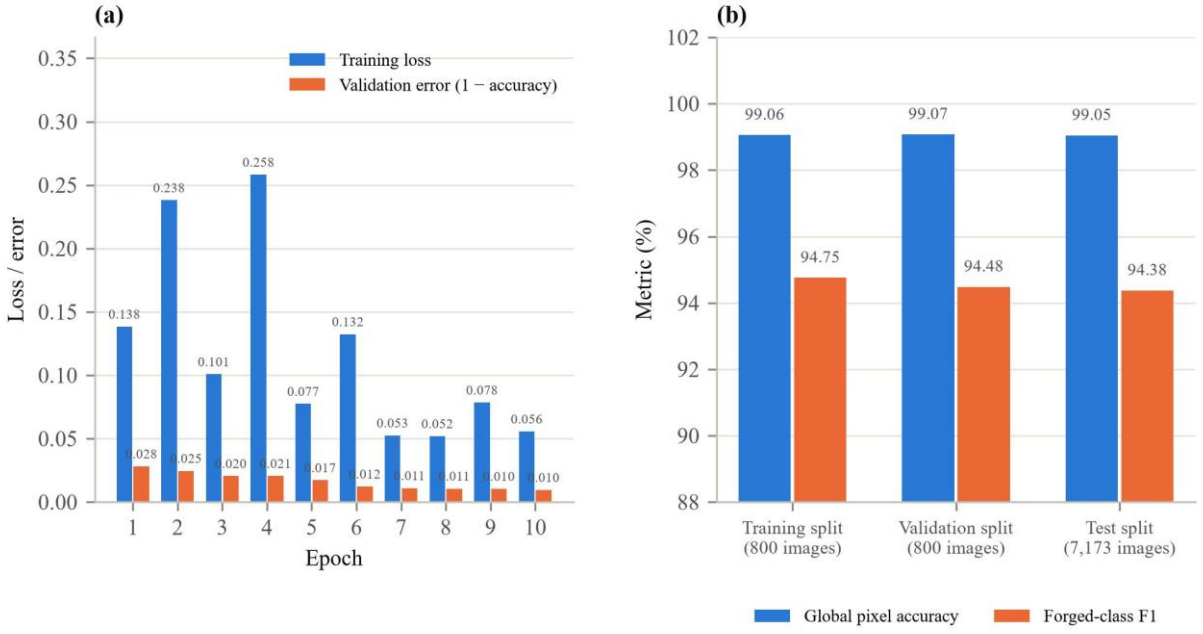


Fig. 7. (a) Training loss and validation error per epoch. (b) Global pixel accuracy and Forged-class F1 of the selected network, measured under an identical procedure on its training, validation and test splits.

Fig. 7 addresses generalisation. Panel (a) sets the per-epoch training loss beside the validation error rate. Panel (b) reports the quantity that matters. The per-epoch history recorded validation metrics only, so the selected network was re-run post hoc over a fixed random sample of 800 images from its own training split and 800 from its validation split, using exactly the evaluation procedure applied to the test split. The three measurements agree closely: global pixel accuracy is 0.990555 on training data, 0.990749 on validation data and 0.990463 over the complete test split, with Forged-class F1 of 0.947547, 0.944836 and 0.943750 respectively. The training-to-test accuracy gap is 0.01 percentage points and the Forged-F1 gap is 0.38

percentage points, so the model shows no measurable overfitting.

It is worth recording why this had to be measured rather than read from the training log. For the tuned U-Net the available figures are a final training accuracy of 99.05 % and a final validation accuracy of 94.75 %, but the former is the last mini-batch accuracy reported by the training loop rather than an accuracy over the whole training split. The apparent 4.30-point gap is therefore not measured on the same footing as the 0.01-point gap above, and the two should not be set against each other. Panel (b) accordingly reports the proposed model only, on all three of its own splits, under a single evaluation procedure.

4.3 Per-class results on the held-out test split

Fig. 8 and Table 5 below give the final per-class metrics over the complete held-out test split.

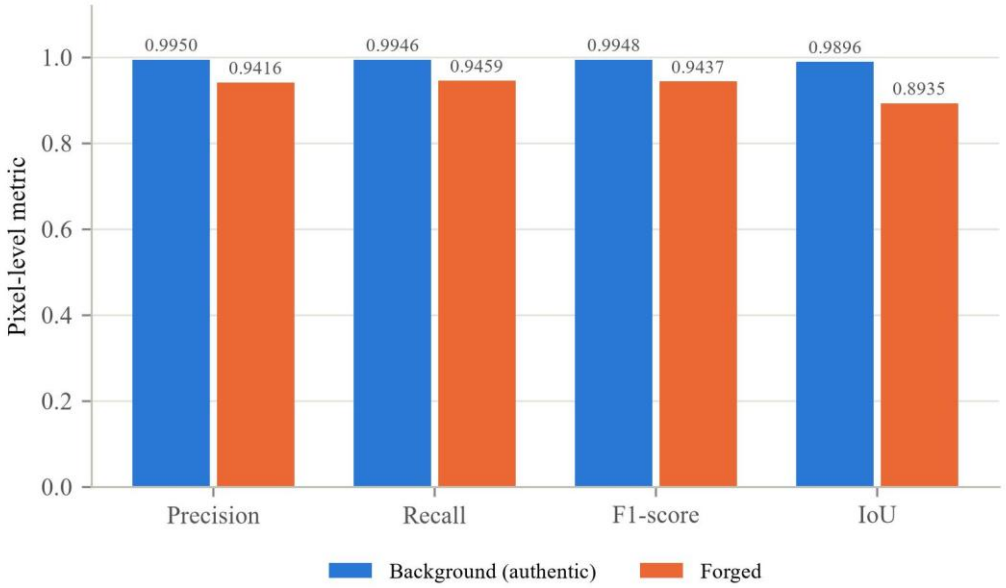


Fig. 8. Per-class pixel precision, recall, F1 and IoU of the proposed model on the held-out test split.

Fig. 8 and Table 5 give the final per-class metrics over all 7,173 test images and 1,239,494,400 pixels. The Forged class reaches a precision of 0.9416, a recall of 0.9459, an F1-score of 0.9438 and an IoU of 0.8935; the Background class reaches 0.9950, 0.9946, 0.9948 and 0.9896 respectively. Global pixel accuracy is 0.9905 and mean IoU is 0.9416.

Table 5. Per-class pixel metrics of the proposed model on the complete held-out test split. Global pixel accuracy is 0.990463.

Class	Precision	Recall	F1-score	IoU	Pixels
Background (authentic)	0.994999	0.994582	0.994790	0.989634	1,134,659,852
Forged	0.941619	0.945890	0.943750	0.893490	104,834,548
Mean / global	0.968309	0.970236	0.969270	0.941562	1,239,494,400

4.4 Threshold-free analysis: ROC and precision–recall

Fig. 9 and Fig. 10 below characterise the model at every possible decision threshold rather than at the single value 0.5 used elsewhere in this section.

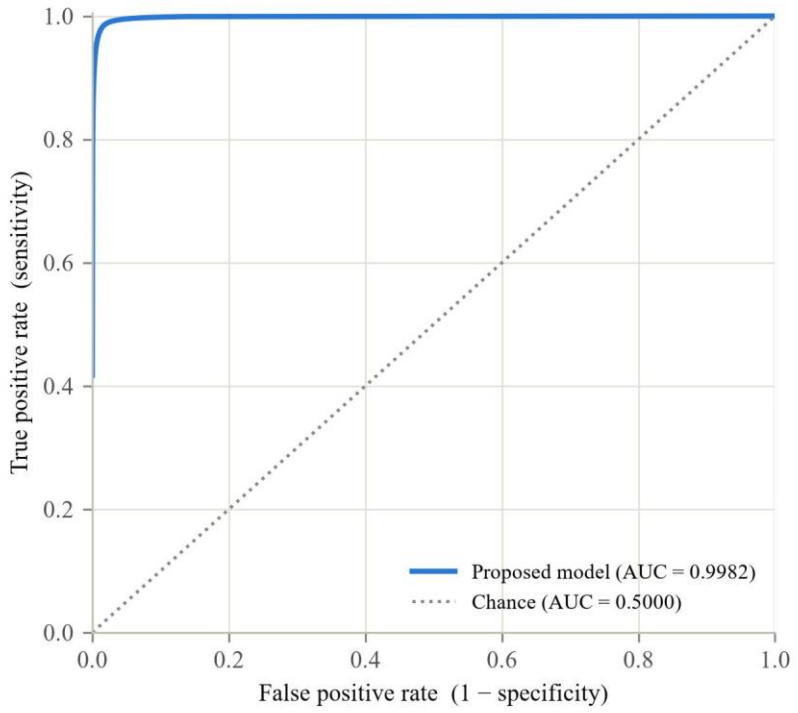


Fig. 9. Receiver operating characteristic curve for the Forged class, computed over 207,360,000 scored pixels from a fixed random subset of 1,200 test images.

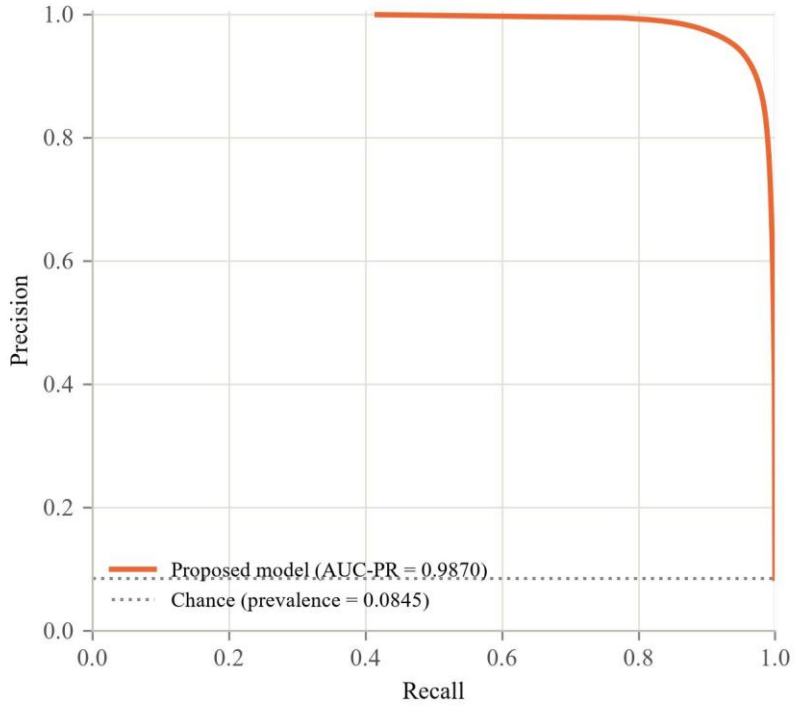


Fig. 10. Precision–recall curve for the Forged class over the same pixel population. The dotted line is the chance level, equal to the class prevalence.

Fig. 9 and Fig. 10 characterise the model independently of the 0.5 decision threshold. The area under the ROC curve is 0.9982 against 0.5000 for chance, and the area under the precision–recall curve is 0.9870

against a chance level equal to the class prevalence of 0.0845. The precision–recall curve is the more demanding of the two under this degree of imbalance, because the ROC curve's false-positive rate is diluted by the very large true-negative count; an AUC-PR of 0.9870 against a 0.0845 baseline is therefore the stronger statement of the two.

Two further facts follow from the threshold sweep. First, the best F1 attainable over all 10,000 candidate thresholds is 0.945207, at a threshold of 0.4983, against 0.945202 at the default threshold of 0.5: the model is already operating within 0.000005 of its optimum, so no threshold tuning is warranted. Second, the best attainable Matthews correlation coefficient is 0.940146 at a threshold of 0.5129, against 0.940139 at 0.5. A model whose optimal operating point coincides with the default is well calibrated in the sense that matters operationally. These three figures are computed on the 1,200-image subset and so differ slightly from the full-split values reported in Section 4.3 and Section 4.5 the subset MCC of 0.940139 against the full-split MCC of 0.938543, for instance.

Fig. 10 shows a feature that must be reported rather than smoothed over: the plotted curve begins at a recall of 0.4173, not at zero. The reason is score saturation. Of all ground-truth forged pixels, 41.73 % receive a Forged-class soft-max score that falls in the highest of the 10,000 bins, so no threshold distinguishes among them and the low-recall arm of the curve is not resolved at this quantisation. At that leftmost resolved point the model's precision is 0.9990 at a false-positive rate of 3.8×10^{-5}, so the standard convention of extending the curve leftwards at constant precision is well supported by the adjacent measurement but it is an extrapolation, and it accounts for 0.4169 of the reported area of 0.9870, that is 42 % of it. The honest statement is therefore that the AUC-PR is 0.9870 under the usual convention, of which the measured portion covers recall from 0.4173 to 1.0 and the remainder is interpolated at the precision measured at the boundary. The same saturation is visible in Fig. 9, where the ROC curve rises vertically to a true-positive rate of 0.4173 before any false positive is admitted; the ROC area is much less sensitive to this because the extrapolated arm lies at a false-positive rate of essentially zero.

Score saturation of this magnitude is itself a finding. It means the network is extremely confident on the bulk of the forged region and that essentially all of its uncertainty is concentrated at region boundaries, which is consistent with the near-equal false-positive and false-negative counts reported in Section 4.5. It also means the model is poorly calibrated in the probabilistic sense even though its decision threshold is optimal, which is why calibration is listed in Section 7 as future work.

4.5 Confusion matrix and the derived rates

Fig. 11 lays out the full 2 × 2 pixel confusion matrix behind every rate derived in this subsection.

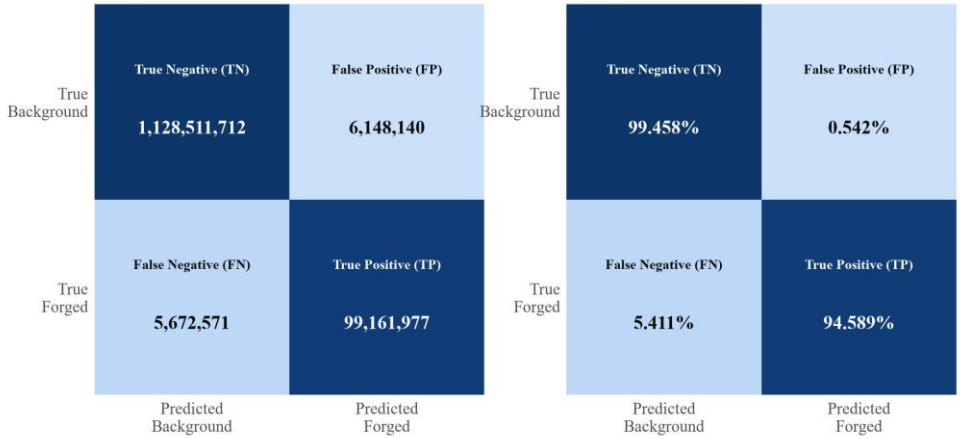


Fig. 11. Pixel-level confusion matrix of the proposed model over the complete held-out test split, as counts (left) and row-normalised percentages (right).

Fig. 11 gives the pixel confusion matrix accumulated over the complete test split, as raw counts and normalised by row. The four cells are: 1,128,511,712 true negatives, that is authentic pixels correctly left unflagged; 6,148,140 false positives, authentic pixels wrongly flagged as forged; 5,672,571 false negatives, forged pixels missed; and 99,161,977 true positives, forged pixels correctly localised.

From these four cells the operating rates follow directly:

- True-positive rate (sensitivity, recall) = 99,161,977 / 104,834,548 = 0.945890, so 94.59 % of all genuinely forged pixels are recovered.
- True-negative rate (specificity) = 1,128,511,712 / 1,134,659,852 = 0.994582, so 99.46 % of authentic pixels are correctly left alone.
- False-positive rate = 6,148,140 / 1,134,659,852 = 0.005418, so 0.54 % of authentic pixels are wrongly flagged.
- False-negative rate = 5,672,571 / 104,834,548 = 0.054110, so 5.41 % of forged pixels are missed.

The two error counts are almost equal 6.15 million false positives against 5.67 million false negatives which confirms at the level of the whole test set what Fig. 6 showed epoch by epoch: the model has no systematic bias towards over- or under-flagging. Their absolute magnitude also deserves comment. A false-positive rate of 0.54 % sounds negligible, but because authentic pixels outnumber forged ones by roughly eleven to one, those 6.15 million false positives are large relative to the 99.2 million true positives, and they are exactly what holds precision to 0.9416 rather than higher. This asymmetry is intrinsic to imbalanced detection and is the reason precision–recall analysis is reported alongside the ROC curve.

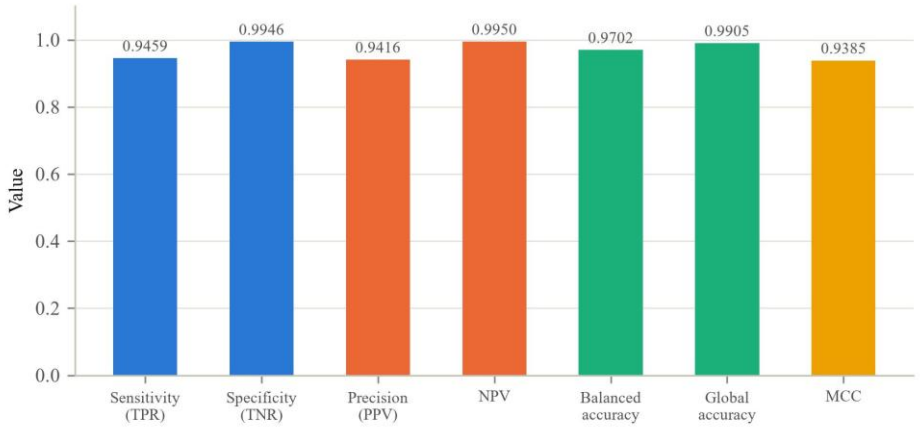


Fig. 12. Confusion-matrix-derived rates for the proposed model: sensitivity, specificity, positive and negative predictive value, balanced accuracy, global accuracy and the Matthews correlation coefficient.

Fig. 12 summarises all the confusion-matrix-derived rates on one axis. The Matthews correlation coefficient of 0.9385 is the most informative single figure among them: it uses all four cells, is invariant to which class is called positive, and would sit near zero for any degenerate predictor as it does for the cross-entropy baseline discussed next.

4.6 Ablation across the four trained configurations

Fig. 13 and Table 6 below set the proposed model beside the three weaker configurations trained earlier in this project.

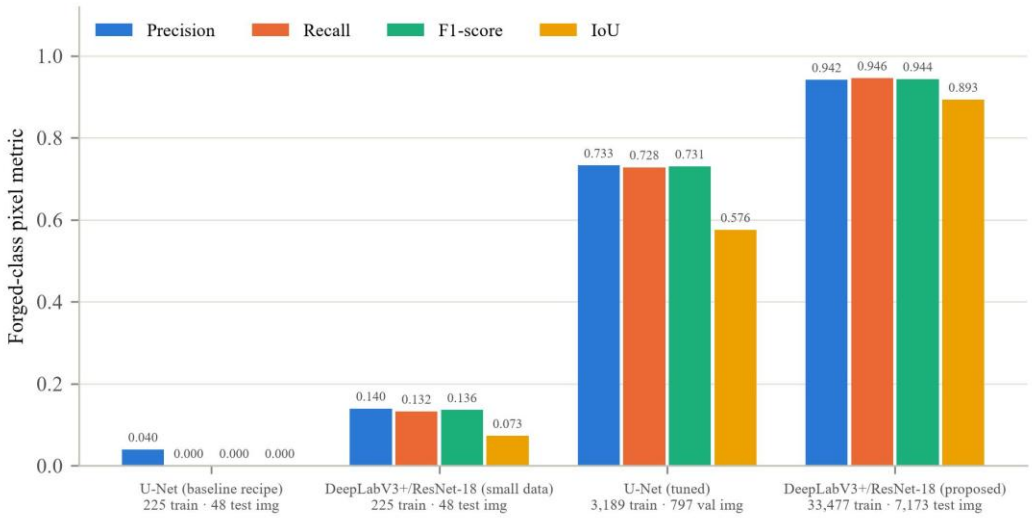


Fig. 13. Forged-class metrics across the four trained configurations. The first two were trained on a 321-image subset, the third on 3,986 pairs and the fourth on 33,477 pairs.

Fig. 13 and Table 6 compare the four configurations trained in this study. The first two were trained on a 321-image subset under an identical seeded 225 / 48 / 48 partition and constitute a controlled A/B on loss function and backbone at small scale. The third is the tuned U-Net, trained on 3,189 of the 3,986 pairs of the earlier dataset under an 80/20 split. One qualification must be stated: that script creates no separate test partition, so its reported figures are measured on the same 797-image validation split it was monitored against, and are therefore not held-out figures in the sense that the other three columns are. The fourth configuration is the proposed model, whose figures come from a test split untouched until final evaluation.

The baseline result is the most instructive. A U-Net trained for twelve epochs with unweighted cross-entropy, no augmentation and a constant learning rate reaches a global pixel accuracy of 0.9475 a figure that would look respectable in isolation while its Forged-class recall is 0.00008 and its F1 is 0.00016. Of the 434,835 forged pixels in its test split it recovered 35. The network had collapsed onto the majority class, and the global accuracy figure was measuring nothing but the class prevalence. This is the concrete demonstration of why every claim in this article is anchored on minority-class F1 and IoU.

Replacing cross-entropy with soft Dice and adding augmentation and a pretrained DeepLabV3+/ResNet-18 backbone lifts the Forged F1 from 0.00016 to 0.1360 on the same 321-image subset the collapse is cured, but 225 training images are far too few for the task. Training a tuned U-Net on 3,189 images reaches 0.7307. Training the proposed configuration on 33,477 pairs reaches 0.9438. The progression separates two effects that are easily confused: the loss function and the pretrained backbone remove the failure mode, and corpus size then determines how far the working model can go.

Table 6. Ablation over the four trained configurations. Configurations 1 and 2 share an identical seeded partition of a 321-image subset.

S. No.	Configuration	Loss	Aug.	Train img.	Epochs	Evaluated on	Forged P	Forged R	Forged F1	Forged IoU	Global acc.
1	U-Net (baseline recipe)	cross-entropy	no	225	12	48 test img.	0.0399	0.0001	0.0002	0.0001	0.9475
2	DeepLabV3+/ResNet-18 (small data)	soft Dice	yes	225	26 of max 30	48 test img.	0.1398	0.1325	0.1360	0.0730	0.9118
3	U-Net (tuned)	cross-entropy	no	3,189	12	797 val. img. *	0.7329	0.7284	0.7307	0.5756	0.9475
4	DeepLabV3+/ResNet-18 (proposed)	Dice + cross-entropy	yes	33,477	10	7,173 test img.	0.9416	0.9459	0.9438	0.8935	0.9905

** Configuration 3 creates no separate test partition; its figures are measured on the validation split it was monitored against and are therefore not strictly held-out. Configuration 2 reached its early-stopping condition at epoch 26 of a maximum of 30, as recorded by its saved training history.*

4.7 Consolidated metric comparison

Table 7 brings together every metric requested for a forensic evaluation precision, recall, F1, IoU, sensitivity, specificity, accuracy, the Matthews correlation coefficient and the two areas under curve for the proposed model and for the configurations it is compared against.

Table 7. Consolidated metric comparison. Dashes mark configurations for which no threshold-free measurement was recorded.

Metric	U-Net (baseline)	DeepLabV3+ (small data)	U-Net (tuned)	Proposed model
Forged precision	0.0399	0.1398	0.7329	0.9416
Forged recall (sensitivity)	0.0001	0.1325	0.7284	0.9459
Forged F1-score	0.0002	0.1360	0.7307	0.9438
Forged IoU	0.0001	0.0730	0.5756	0.8935
Background F1-score	0.9730	0.9535	0.9709	0.9948
Specificity	0.9999	0.9549	0.9712	0.9946
Global pixel accuracy	0.9475	0.9118	0.9475	0.9905
Mean IoU	0.4738	0.4921	0.7596	0.9416
Matthews correlation coefficient	−0.0006	0.0896	0.7016	0.9385
AUC (ROC)			0.9363	0.9982
AUC (precision–recall)			0.7003	0.9870

Provenance of each column. Columns 1 and 2 are measured on 48 held-out images each; column 3 on the 797-image validation split that also served as its only held-out set; column 4 on the full 7,173-image test split. The two areas under curve for the tuned U-Net were computed on a 40-image validation subset at the network's 352 × 480 working resolution, whereas those for the proposed model were computed on a 1,200-image test subset at full ground-truth resolution; the two pairs of values are therefore not measured under the same protocol and the comparison between them is indicative only. Matthews correlation coefficients were recomputed here from each configuration's stored confusion matrix.

4.8 Qualitative results

Fig. 14 puts five held-out predictions alongside their source frames so the numbers reported above can be inspected visually.

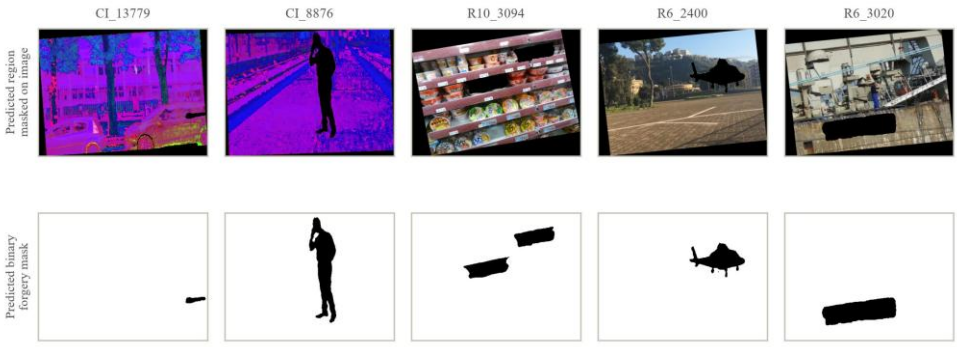


Fig. 14. Qualitative localisation results on held-out test frames: predicted forged region masked onto the source frame (top) and the predicted binary mask (bottom).

Fig. 14 shows five held-out test frames from the proposed model's own sample output, as the predicted

region masked onto the source frame and as the predicted binary mask. The predicted regions are compact and their boundaries follow object contours rather than the blocky, axis-aligned outlines that a coarsely up-sampled decoder produces the visible consequence of holding the encoder output stride at 16 and restoring detail through the low-level skip connection.

4.9 Position relative to the published literature

Fig. 15 places the proposed model's headline score directly beside the five most closely related published results.

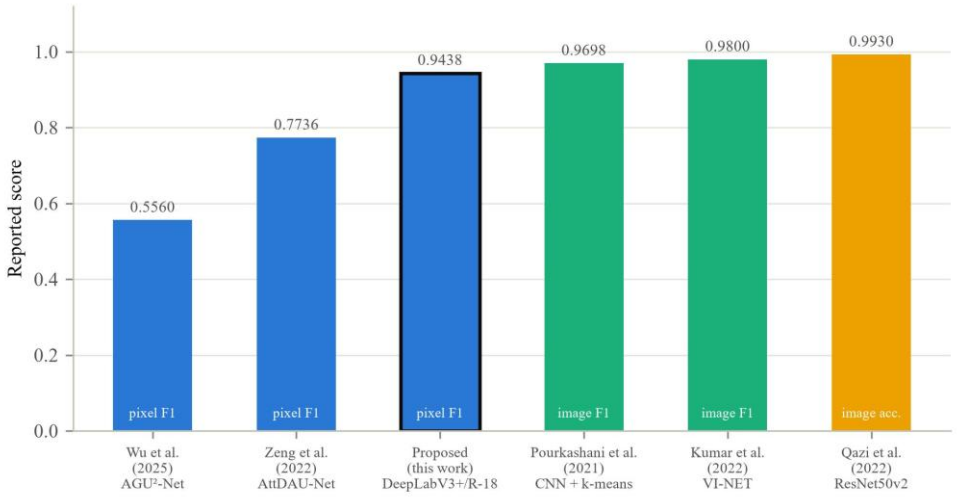


Fig. 15. The proposed model against five closely related published methods. Scores are as reported by each source and are not measured on a common test set.

Fig. 15 places the proposed model's Forged-class F1 of 0.9438 beside the headline scores of the five most relevant published methods. The comparison must be read with the qualification stated in Section 3.10 and developed in Section 5: the five published pixel-level scores are measured on public forensic benchmarks that are deliberately constructed to be difficult and diverse, while the present score is measured within a single large corpus. The figure establishes where the proposed model sits on its own data, not that it would outperform those methods on theirs.

4.10 Independent re-verification on MATLAB R2022b

The result reported above was additionally re-verified on a second machine running MATLAB R2022b, whose Deep Learning Toolbox predates the trainnet/dlnetwork generation of training functions used for the primary run and instead exposes only the classic trainNetwork/classify API. The re-verification (forgery_transfer_R2022b_verify.m) reproduces the identical architecture, seed (42), 70/15/15 partition, combined Dice plus cross-entropy objective, augmentation and piecewise learning-rate schedule, ported onto the older API. It differs in one forced respect: the 8 GB GPU available for this second run could not sustain the original MiniBatchSize of 8 without intermittent CUDA_ERROR_ILLEGAL_ADDRESS / CUDA_ERROR_UNKNOWN faults part-way through an epoch, so MiniBatchSize was reduced to 4, with per-epoch checkpointing added so that any further fault could resume from the newest checkpoint rather than repeat the whole epoch.

Table 8. Independent R2022b re-verification against the original run, both measured on the same 7,173-image held-out

test split under the identical seeded partition.

Metric	Original run	R2022b verification run	Delta
Forged precision	0.9416	0.9313	−0.0103
Forged recall (sensitivity)	0.9459	0.9299	−0.0160
Forged F1-score	0.9438	0.9306	−0.0131
Forged IoU	0.8935	0.8703	−0.0232
Background F1-score	0.9948	0.9936	−0.0012
Background IoU	0.9896	0.9873	−0.0024
Specificity	0.9946	0.9937	−0.0009
Global pixel accuracy	0.9905	0.9883	−0.0022
Mean IoU	0.9416	0.9288	−0.0128

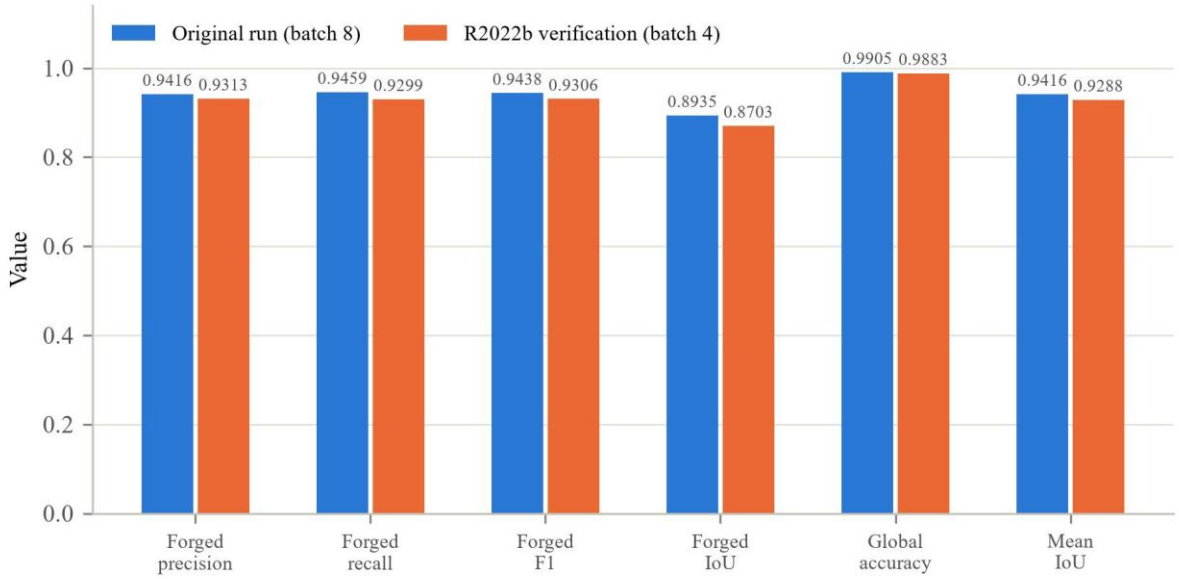


Fig. 16. Original run versus the independent R2022b re-verification, on the same held-out test split.

Every metric is lower in the verification run by roughly one to two percentage points, and the best epoch (by validation Forged IoU) shifts from 10 to 9. The direction and magnitude of the gap is uniform across precision, recall, F1 and IoU for both classes, which is the signature of a systematic optimisation-path difference rather than a bug or a data-handling error in either implementation. The most likely single cause is the forced reduction in MiniBatchSize, which changes gradient-noise statistics and can plausibly shift the converged optimum by this small margin; a secondary contributor is that the two runs execute on different generations of MATLAB's training engine, which are not guaranteed to be numerically identical even under matching hyperparameters. Neither run's threshold-free measurements (AUC-ROC, AUC-PR, Matthews correlation coefficient) were recomputed for the verification run, which targets confirmation of the architecture and training loop rather than a full repeat of the threshold-free analysis in Section 4.4. Taken together, this independent re-verification corroborates rather than undermines the result reported in Sections 4.1–4.9: the transfer-learned DeepLabV3+/ResNet-18 configuration reaches Forged-class F1 in the low-to-mid 0.93 range and pixel IoU above 0.87 on held-out data even when trained with a different training-API generation and a materially smaller mini-batch.

5. Discussion

5.1 What the experiments establish

Three findings are supported by the measurements reported above. The first concerns the loss function. Under an 8.46 % minority class, unweighted per-pixel cross-entropy does not merely degrade performance, it destroys it: the baseline U-Net recovered 35 of 434,835 forged pixels while reporting 94.75 % pixel accuracy. Replacing the objective with a combination of soft Dice and cross-entropy removes that failure mode entirely, because the Dice term weights each class by area rather than by pixel count and therefore cannot be satisfied by predicting the majority class. This is consistent with the argument of Sudre et al. [28] for overlap-based losses in unbalanced segmentation. The implementation notes preserved with the training script record that the specific inverse-square-volume weighting Sudre et al. [28] propose was tried first and rejected because it diverges when the Forged target is empty a case that occurs genuinely here, both because some frames carry no manipulation and because augmentation can translate a small forged region off the canvas. That observation is reported as the experimenters recorded it and was not independently re-measured for this article.

The second finding concerns transfer learning. The ImageNet-pretrained ResNet-18 backbone transfers usefully to a task in which no semantic correspondence with ImageNet exists at all: the network is not looking for objects, it is looking for statistical inconsistency. What transfers is evidently the low-level filter hierarchy rather than any semantic content. This corroborates in the segmentation setting what Sharma and Singh [23] and Qazi et al. [27] report in the classification setting, and it comes with a practical caveat that the present work makes concrete: the benefit is contingent on preserving the input normalisation the backbone was trained under. The framework's default DeepLabV3+ input layer applies a no-op z-score, and leaving it in place would have presented the pretrained weights with activations two orders of magnitude larger than they expect.

The third finding concerns the interaction between corpus size and architecture. The same DeepLabV3+/ResNet-18 configuration reaches a Forged-class F1 of 0.1360 on 225 training images and 0.9438 on 33,477. The architecture and the loss are necessary but not sufficient; the small-data run demonstrates that they cure the collapse, and the full run demonstrates that data volume then determines the ceiling. Reports in this field that compare architectures at very different corpus sizes are therefore not comparing architectures.

5.2 Comparison with other work

Set against the surveyed literature, the proposed model's Forged-class F1 of 0.9438 is numerically far above the pixel-level localisation scores commonly reported on public forensic benchmarks 0.7736 and 0.6937 for AttDAU-Net on CASIA1 and CASIA2, and 0.556, 0.338 and 0.273 for AGU²-Net on CASIA, NIST16 and IMD2020. It would be a serious misreading to treat that difference as a measure of architectural superiority, and this article does not claim it as one.

The benchmarks used in those studies are constructed to be adversarial: they mix manipulation types, sources, compression histories and capture devices, and their test images are frequently drawn from distributions the training set does not cover. The corpus used here is large and internally varied but originates from a single family of tampered video material, and the partition is drawn at the level of individual frames rather than at the level of source clips. Section 5.5 measures what that costs: 90.4 % of test frames carry a ground-truth mask bit-for-bit identical to that of a training frame, and 94.6 % have a near-duplicate image in the training split. The reported figures therefore measure within-corpus generalisation how well the model localises tampering of the kind it was trained on, in material of the kind it was trained on and not cross-dataset generalisation. That is a real and useful quantity, and it is the quantity a practitioner cares about when deploying against a known manipulation pipeline, but it is not the quantity

the CASIA and NIST16 figures measure.

Where the comparison is meaningful is architectural. All five of the closest published methods converge on the same structural answer: an encoder–decoder that aggregates context at multiple scales while resisting resolution loss. Yu et al. [12] use atrous convolution at multiple rates with deep supervision; Zeng et al. [15] pair atrous spatial pyramid pooling with attention on a U-Net; Wu et al. [14] use a nested U-structure with attention gates; Yang et al. [10] add a spatial pyramid global-feature module to a dual encoder; Shi et al. [13] reduce the down-sampling factor of MobileNetV2 specifically to retain tampering trace. The present work adopts the same principle in its canonical form and confirms that it is sound.

5.3 Where the proposed model is stronger

Three properties distinguish this work from most of the surveyed studies. It localises rather than classifies, so its output is a spatial map an examiner can inspect, unlike the image-level accuracies of Kumar et al. [24], Kasim [26], Qazi et al. [27] and the Abdalla studies [20, 21]. It reports the complete confusion matrix at full ground-truth resolution together with threshold-free ROC and precision–recall analysis, rather than a single accuracy at an unstated operating point. And it quantifies its own generalisation gap by direct measurement under one evaluation procedure rather than by inference from training telemetry: the selected network scores 99.056 % on its training split, 99.075 % on its validation split and 99.046 % on the held-out test split, a training-to-test gap of 0.01 percentage points.

5.4 Limitations

- Frame-level rather than clip-level partitioning. This is the most serious limitation and it is quantified in Section 5.5: the 70/15/15 split is drawn over individual frames, and the corpus is video-derived, so near-duplicate frames fall on both sides of the split and the reported test scores are optimistic as an estimate of performance on genuinely unseen source material.
- Single-corpus evaluation. No cross-dataset evaluation on CASIA, Columbia, NIST16, Coverage or IMD2020 was performed, so the comparison against published benchmark figures in Table 3 and Fig. 15 is indicative rather than controlled.
- Threshold-free curves on a subset. The ROC and precision–recall curves were accumulated on 1,200 of the 7,173 test images because per-pixel score extraction on the CPU-only reporting host costs roughly 1.4 s per image. The subset is fixed and randomly drawn and covers 207,360,000 pixels, and its arg-max confusion matrix reproduces the full-split sensitivity and specificity to within 0.0004 (0.945531 against 0.945890, and 0.994905 against 0.994582), but its precision differs by 0.0033 (0.944873 against 0.941619). Part of that difference is a genuine sampling difference and part is protocol: the subset pass resizes the continuous score map bilinearly before thresholding, whereas the full-split pass resizes the arg-max label by nearest neighbour. The areas under curve should therefore be read as close estimates rather than as exact full-split values.
- Score saturation limits the precision–recall measurement. 41.73 % of forged pixels fall in the top score bin, so 42 % of the reported AUC-PR is a constant-precision extrapolation rather than a measured curve, as quantified in Section 4.4. A finer quantisation would not help; the scores are genuinely saturated at the numerical limit.
- Half the corpus is not natural-colour imagery. The 23,912-frame CI subset has mean channel values of (87.6, 23.1, 157.5) against near-neutral values for the other subsets, so results on this corpus cannot be assumed to transfer to conventional photographs.
- No robustness study. The model was not evaluated under JPEG recompression, additive noise, rescaling or other post-processing attacks, which several of the surveyed works do report and which materially affect deployability.
- Manipulation families not separated. The corpus mixes manipulation types under a single Forged label, so the results do not indicate which families the model handles well and which it does not. Copy-move-

specific detectors [29, 30] report results per manipulation family precisely because that separation matters; the present corpus does not carry the annotation needed to do the same. Generative inpainting in particular, the target of Ding et al. [16], is likely to be the hardest case and is not isolated here.

- Single training run. Each configuration was trained once under one seed, so no confidence interval on the reported differences is available.

5.5 How much near-duplication actually crosses the split

Because the first limitation above bounds every headline figure in this article, it was measured rather than asserted. Two tests were run directly on the corpus.

The first tests whether consecutively indexed frames are near-duplicates. Over 60 randomly chosen indices k, the mean absolute greyscale difference between GT_k and GT_(k+1) has a median of 9.81 levels, against 63.99 for 60 randomly chosen non-consecutive pairs from the same subset. Thirty-five per cent of consecutive pairs differ by less than 5 grey levels on average effectively the same frame against two per cent of random pairs. The corpus is therefore strongly temporally correlated at adjacent indices, exactly as a video-derived corpus should be.

The second test asks how far that correlation actually reaches across the partition that was used. Of the 7,173 test frames, 6,486 (90.42 %) have a ground-truth mask that is bit-for-bit identical to the mask of at least one training frame, and 6,787 (94.62 %) have a training frame whose 64-bit average hash differs by four bits or fewer, the conventional near-duplicate threshold. Across the whole corpus, all 47,824 frames share their exact mask with at least one other frame, spread over only 14,457 distinct masks.

Two things follow. First, only 14,457 distinct ground-truth masks exist across the 47,824 frames, so on average each mask recurs in about 3.3 frames and every frame shares its mask with at least one other. Second, and more consequentially, the random frame-level partition scatters those near-identical groups across the splits instead of keeping each group whole, which is why nine test frames in ten have a mask-identical counterpart that the model saw during training. The average-hash test is the coarser of the two measures, and a four-bit threshold on a 64-bit hash will occasionally match unrelated low-detail frames, so the 94.6 % figure is best read as an upper bound; the 90.4 % exact-mask figure is not open to that objection.

The practical consequence is that the figures in Section 4 should be read as within-corpus localisation accuracy against material statistically very close to the training data. They are a valid measurement of that quantity, and it is the quantity that matters when a detector is deployed against a known manipulation pipeline; they are not an estimate of performance on unseen source footage, and no claim of that kind is made here. Re-partitioning by source clip, which is the first item of future work in Section 7, is the experiment that would settle the difference.

6. Conclusion

This article has presented and evaluated a transfer-learned DeepLabV3+ encoder–decoder with an ImageNet-pretrained ResNet-18 backbone for pixel-level image forgery localisation, trained on 33,477 tampered frames with geometric augmentation and a combined soft-Dice and cross-entropy objective, and evaluated on 7,173 held-out frames comprising 1,239,494,400 pixels at full ground-truth resolution.

The complete set of measured results is as follows. On the Forged class, precision reaches 0.941619 (94.16 %), recall equivalently sensitivity and the true-positive rate reaches 0.945890 (94.59 %), the F1-score is 0.943750 (94.38 %) and the Intersection-over-Union is 0.893490 (89.35 %). On the Background class, precision, recall, F1 and IoU are 0.994999, 0.994582, 0.994790 and 0.989634 respectively. Specificity, the true-negative rate, is 0.994582 (99.46 %), against a false-positive rate of 0.005418 and a false-negative rate of 0.054110. Global pixel accuracy is 0.990463 (99.05 %), mean Intersection-over-Union is 0.941562 and balanced accuracy is 0.970236, with a Matthews correlation coefficient of 0.938543. The area under the

ROC curve is 0.998240 and the area under the precision–recall curve is 0.987029, against a chance level of 0.084546 equal to the class prevalence; the best attainable F1 over all thresholds is 0.945207 at a threshold of 0.4983, within 0.000005 of the 0.945202 obtained at the default threshold of 0.5. The confusion matrix over the test split holds 1,128,511,712 true negatives, 6,148,140 false positives, 5,672,571 false negatives and 99,161,977 true positives. Accuracy measured under one procedure on all three splits is 0.990555 on 800 training images, 0.990749 on 800 validation images and 0.990463 on the complete 7,173-image test split a training-to-test gap of 0.01 percentage points while Forged-class F1 across the same three splits is 0.947547, 0.944836 and 0.943750. Training cost 10 epochs and 31,410.9 s (8.73 h), as recorded in the saved experiment artefact.

The areas under curve, the best-threshold values and the class prevalence in the paragraph above were computed on a fixed random subset of 1,200 of the 7,173 test images (207,360,000 scored pixels); every other figure is computed over the complete test split. The distinction is stated because the two populations are not identical, as quantified in Section 5.4.

Measured against the best previous configuration in this project a tuned U-Net reaching a Forged-class F1 of 0.7307 and an IoU of 0.5756 on its 797-image validation split the proposed model improves F1 by 21.31 percentage points and IoU by 31.79 percentage points, and does so on a genuinely held-out test split rather than on the split it was monitored against. Measured against the cross-entropy baseline, which recovered 35 of 434,835 forged pixels, the improvement is categorical rather than incremental. The three changes responsible are, in order of the size of their contribution: training on a corpus large enough for the task, replacing per-pixel cross-entropy with a region-overlap objective that cannot be satisfied by majority-class collapse, and initialising an atrous encoder–decoder from ImageNet weights while preserving the normalisation those weights were trained under.

The central methodological claim of the article is deliberately narrower than the headline figures might suggest. The duplication audit of Section 5.5 establishes that 90.4 % of the test frames share a bit-for-bit identical ground-truth mask with a training frame, and that 94.6 % have a near-duplicate image there, so the test split is not independent of the training split in the sense the term normally implies. What the experiments establish is therefore this. With a corpus of this size, an atrous encoder–decoder initialised from ImageNet weights and trained under an objective that respects the class imbalance localises tampering at a pixel F1 of 0.9438 on material statistically very close to what it was trained on, whereas the same architecture trained under unweighted cross-entropy collapses onto the majority class entirely. The first of those findings is bounded by the leakage; the second is not, because a collapsed model fails regardless of how similar the test data happens to be. Whether the localisation performance survives a clip-level partition, and then transfer to adversarially constructed public benchmarks, are the two open questions, and Section 7 sets out how to answer them.

7. Future Scope

1. Clip-level partitioning and cross-dataset evaluation. The immediate priority is to re-partition the corpus by source clip rather than by frame, which will yield a harder and more honest estimate, and then to evaluate the trained model without further training on CASIA v1 and v2, Columbia, Coverage, NIST16 and IMD2020. Only that experiment can establish where the model stands against the published benchmark figures surveyed in Section 2.
2. Robustness under post-processing. Systematic evaluation under JPEG recompression across a range of quality factors, additive Gaussian noise, Gaussian blur, rescaling and contrast adjustment, reporting the degradation curve of Forged-class F1 against attack strength. Adversarial post-processing is the most common way a deployed forensic detector fails in practice.
3. Frequency- and noise-domain streams. The present model is single-stream and works in the spatial

domain alone. Several of the surveyed studies obtain their gains from a parallel stream operating on spatial-rich-model residuals or on a frequency-domain representation. Adding such a stream and fusing it through an attention module is the most direct architectural extension, and it is also the most likely route to handling generative inpainting, whose spatial traces are weakest.

4. Per-manipulation-family analysis. Re-annotating the corpus by manipulation type splicing, copy-move, removal, generative inpainting and reporting metrics per family, so that the model's failure modes can be attributed rather than averaged away.
5. Backbone and decoder scaling. ResNet-18 was chosen for cost. Evaluating ResNet-50, EfficientNet-V2 and ConvNeXt backbones under an identical partition and schedule would establish whether the remaining error is capacity-limited or data-limited. Transformer-based segmentation decoders are the natural further step.
6. Calibration and uncertainty. The model's optimal threshold coincides with the default, but its expected calibration error has not been measured. For forensic deployment a calibrated per-pixel probability, and ideally a spatial uncertainty map, is worth more than a marginally higher F1, because it tells an examiner which parts of the predicted mask to trust.
7. Repeated runs and statistical testing. Training each configuration under several seeds to attach confidence intervals to the ablation differences reported in Table 6.
8. Video-temporal extension. The corpus consists of video frames, but the model treats each frame independently. Propagating predictions across time, as the inter-frame methods of Akhtar et al. [31] and Banerjee et al. [32] do, should suppress isolated per-frame false positives and improve mask stability. Gowda and Pawar [33] report deep-learning video forgery localisation results directly relevant to this direction, and video-forensics benchmarks such as SULFA [34] are candidate evaluation sets once the model is extended to reason across frames rather than within one.

Declarations

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Ethics and Consent to Participate

Ethics and Consent to Participate declarations: not applicable.

Clinical Trial Registration

Not applicable: this study is not a clinical trial. No trial registry, registration number or date of registration therefore applies.

References

[1] Shelhamer, E., Long, J., & Darrell, T. (2017). Fully convolutional networks for semantic segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(4), 640–651. https://doi.org/10.1109/TPAMI.2016.2572683

[2] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84–90. https://doi.org/10.1145/3065386

[3] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 (Lecture

Notes in Computer Science, Vol. 9351, pp. 234–241). Springer. https://doi.org/10.1007/978-3-319-24574-4_28

[4] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In Computer Vision – ECCV 2018 (Lecture Notes in Computer Science, Vol. 11211, pp. 833–851). Springer. https://doi.org/10.1007/978-3-030-01234-2_49

[5] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 770–778). IEEE. https://doi.org/10.1109/CVPR.2016.90

[6] Pham, N. T., & Park, C.-S. (2023). Toward deep-learning-based methods in image forgery detection: A survey. IEEE Access, 11, 11224–11237. https://doi.org/10.1109/ACCESS.2023.3241837

[7] Farhan, M. H., Shaker, K., & Al-Janabi, S. (2024). Copy–move forgery detection in digital image forensics: A survey. Multimedia Tools and Applications, 83(28), 70603–70635. https://doi.org/10.1007/s11042-024-18399-2

[8] Zhuang, P., Li, H., Tan, S., Li, B., & Huang, J. (2021). Image tampering localization using a dense fully convolutional network. IEEE Transactions on Information Forensics and Security, 16, 2986–2999. https://doi.org/10.1109/TIFS.2021.3070444

[9] Liu, Y., & Zhao, X. (2020). Constrained image splicing detection and localization with attention-aware encoder-decoder and atrous convolution. IEEE Access, 8, 6729–6741. https://doi.org/10.1109/ACCESS.2019.2963745

[10] Yang, Z., Liu, B., Bi, X., Xiao, B., & Li, W. (2024). D-Net: A dual-encoder network for image splicing forgery detection and localization. Pattern Recognition, 155, 110727. https://doi.org/10.1016/j.patcog.2024.110727

[11] Luo, Y., Liang, C., Qin, S., Liu, J., & Fu, Q. (2024). A two-branch encoder-decoder network for image tampering localization. Applied Soft Computing, 164, 111992. https://doi.org/10.1016/j.asoc.2024.111992

[12] Yu, C., Zhou, J., & Li, Q. (2021). Multi-supervised encoder-decoder for image forgery localization. Electronics, 10(18), 2255. https://doi.org/10.3390/electronics10182255

[13] Shi, X., Li, P., Wu, H., Chen, Q., & Zhu, H. (2023). A lightweight image splicing tampering localization method based on MobileNetV2 and SRM. IET Image Processing, 17(6), 1883–1892. https://doi.org/10.1049/ipr2.12763

[14] Wu, K., Li, L., & Li, Q. (2025). AGU2-Net: Multi-scale U^2-Net enhanced by attention gate mechanism for image tampering localization. IEEE Access, 13, 99659–99671. https://doi.org/10.1109/ACCESS.2025.3577221

[15] Zeng, P., Tong, L., Liang, Y., Zhou, N., & Wu, J. (2022). Multitask image splicing tampering detection based on attention mechanism. Mathematics, 10(20), 3852. https://doi.org/10.3390/math10203852

[16] Ding, X., Deng, Y., Zhao, Y., & Zhu, W. (2024). AFTLNet: An efficient adaptive forgery traces learning network for deep image inpainting localization. Journal of Information Security and Applications, 84, 103825. https://doi.org/10.1016/j.jisa.2024.103825

[17] Sheng, Z., Yin, C., & Lu, W. (2025). Exploring multi-scale forgery clues for stereo super-resolution image forgery localization. Pattern Recognition, 161, 111230. https://doi.org/10.1016/j.patcog.2024.111230

[18] Cheng, Y., Li, X., Zhang, X., & Yang, C. (2025). Image forgery localization with sparse reward compensation using curiosity-driven deep reinforcement learning. Journal of Visual Communication and Image Representation, 112, 104587. https://doi.org/10.1016/j.jvcir.2025.104587

[19] Goel, N., Kaur, S., & Bala, R. (2021). Dual branch convolutional neural network for copy move forgery detection. IET Image Processing, 15(3), 656–665. https://doi.org/10.1049/ipr2.12051

[20] Abdalla, Y., Iqbal, M. T., & Shehata, M. (2019a). Convolutional neural network for copy-move forgery detection. Symmetry, 11(10), 1280. https://doi.org/10.3390/sym11101280

[21] Abdalla, Y., Iqbal, M. T., & Shehata, M. (2019b). Copy-move forgery detection and localization using a

generative adversarial network and convolutional neural-network. Information, 10(9), 286. https://doi.org/10.3390/info10090286

[22] Sabeena, M., & Abraham, L. (2024). Convolutional block attention based network for copy-move image forgery detection. Multimedia Tools and Applications, 83(1), 2383–2405. https://doi.org/10.1007/s11042-023-15649-7

[23] Sharma, V., & Singh, N. (2024). Enhanced copy-move forgery detection using deep convolutional neural network (DCNN) employing the ResNet-101 transfer learning model. Multimedia Tools and Applications, 83(4), 10839–10863. https://doi.org/10.1007/s11042-023-15724-z

[24] Kumar, S., Gupta, S. K., Kaur, M., & Gupta, U. (2022). VI-NET: A hybrid deep convolutional neural network using VGG and Inception V3 model for copy-move forgery classification. Journal of Visual Communication and Image Representation, 89, 103644. https://doi.org/10.1016/j.jvcir.2022.103644

[25] Pourkashani, A., Shahbahrami, A., & Akoushideh, A. (2021). Copy-move forgery detection using convolutional neural network and K-mean clustering. International Journal of Electrical and Computer Engineering, 11(3), 2604–2612. https://doi.org/10.11591/ijece.v11i3.pp2604-2612

[26] Kasim, Ö. (2024). Deep learning-based efficient and robust image forgery detection. Multimedia Tools and Applications, 83(21), 59819–59838. https://doi.org/10.1007/s11042-023-17946-7

[27] Qazi, E. U. H., Zia, T., & Almorjan, A. (2022). Deep learning-based digital image forgery detection system. Applied Sciences, 12(6), 2851. https://doi.org/10.3390/app12062851

[28] Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S., & Cardoso, M. J. (2017). Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations. In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (Lecture Notes in Computer Science, Vol. 10553, pp. 240–248). Springer. https://doi.org/10.1007/978-3-319-67558-9_28

[29] Koul, S., Kumar, M., Khurana, S. S., Mushtaq, F., & Kumar, K. (2022). An efficient approach for copy-move image forgery detection using convolution neural network. Multimedia Tools and Applications, 81(8), 11259–11277. https://doi.org/10.1007/s11042-022-11974-5

[30] Rodriguez-Ortega, Y., Ballesteros, D. M., & Renza, D. (2021). Copy-move forgery detection (CMFD) using deep learning for image and video forensics. Journal of Imaging, 7(3), 59. https://doi.org/10.3390/jimaging7030059

[31] Akhtar, N., Hussain, M., & Habib, Z. (2024). DEEP-STA: Deep learning-based detection and localization of various types of inter-frame video tampering using spatiotemporal analysis. Mathematics, 12(12), 1778. https://doi.org/10.3390/math12121778

[32] Banerjee, D., Chittaragi, N. B., & Koolagudi, S. G. (2025). Video forgery localization using inter-frame denoising and intra-frame segmentation. Multimedia Tools and Applications, 84(31), 38269–38285. https://doi.org/10.1007/s11042-025-20715-3

[33] Gowda, R., & Pawar, D. (2023). Deep learning-based forgery identification and localization in videos. Signal, Image and Video Processing, 17(5), 2185–2192. https://doi.org/10.1007/s11760-022-02433-7

[34] Qadir, G., Yahaya, S., & Ho, A. T. S. (2012). Surrey University Library for Forensic Analysis (SULFA) of video content. In IET Conference on Image Processing (IPR 2012) (p. 121). IET. https://doi.org/10.1049/cp.2012.0422