

Extending a Multi-Model Face-Detection and Feature-Extraction Approach to Video: Temporal-Consistency-Aware Deepfake Detection: A Simulation-Based Evaluation

Willems Aino^{1*}, Darragh Ingrid²

¹ University College Dublin, IRELAND

² Aalto University, FINLAND

Abstract

Background. A deepfake video is not a single still image; it is a sequence of frames whose per-frame manipulation strength a generative method does not necessarily hold constant, and a detector that only ever looks at one frame at a time, or naively averages across frames, can under-use exactly this frame-to-frame inconsistency as a detection signal. Rahman *et al.* investigated face-detection and feature-extraction techniques for deepfake detection using a multi-model machine-learning approach; this article extends that approach, and this study series' companion still-image evaluation of it, from single face crops to video, asking what a frame-level classifier built the same way should do with the resulting sequence of per-frame predictions. **Methods.** Using the strongest detector-and-feature combination identified in the companion still-image evaluation (an MTCNN-style landmark-aligned detector with Xception-style deep-embedding features) and a Random Forest frame-level classifier, this study implements and compares three video-level aggregation strategies: a per-frame majority vote, a mean-probability pool, and an explicit temporal-consistency feature that combines the mean per-frame probability with the frame-to-frame standard deviation of that probability, with every strategy's decision threshold calibrated on training-set videos only. The evaluation uses 1,000 synthetic twelve-frame videos per seed, over five seeds, with fake videos engineered to carry a deliberately weak mean per-frame artifact signal that fluctuates frame to frame, so that no single frame is reliably informative on its own. **Results.** The temporal-consistency feature reached the highest accuracy (0.863), F1 (0.864), and AUC (0.939) of the three strategies, ahead of mean-probability pooling (0.859 / 0.856 / 0.932) and well ahead of majority voting (0.781 / 0.782 / 0.865); it also achieved this with a materially higher recall than mean-probability pooling (0.872 versus 0.835) at a small precision cost (0.856 versus 0.879). **Conclusion.** When a deepfake's artifact signal is weak on any single frame but inconsistent across frames, a design this evaluation argues is a realistic description of how imperfect generative synthesis actually behaves, explicitly modelling that frame-to-frame inconsistency as a detection feature outperforms both naive per-frame aggregation strategies, extending Rahman *et al.*'s multi-model approach usefully from still images to video.

Keywords: Deepfake Detection; Video Forensics; Temporal Consistency; Face Detection; Feature Extraction; Multi-Model Machine Learning; Frame Aggregation; Random Forest

Introduction

A companion article in this study series implemented Rahman *et al.*'s[1] multi-model investigation of face detection and feature extraction for deepfake detection at the level of a single still image, an eighteen-cell grid of face-detection front-ends, feature-extraction representations, and classifiers, and found that feature-extraction quality dominated the other two design axes. That evaluation, like the great majority of still-image deepfake-detection studies, treated every face crop as an independent classification problem. A deepfake video is not well described that way: it is a sequence of frames, generated (in most practical deepfake pipelines) frame by frame or in short overlapping windows, and the manipulation artefacts a detector relies on are not guaranteed, and in practice are not generally observed, to present with constant strength across every frame of a clip. A detector that scores each frame independently and then aggregates those scores is implicitly deciding how to use, or not use, this frame-to-frame variability, whether or not that decision is made deliberately.

This article extends Rahman *et al.*'s multi-model approach, and this study series' own still-image implementation of it, from the single-frame to the video setting. The extension keeps the still-image pipeline's strongest identified components fixed, an MTCNN-style landmark-aligned detector[2], Xception-style deep-embedding features[3], and a Random Forest classifier[4], and asks a narrower, video-specific question: given a sequence of per-frame fake-probability scores from that fixed pipeline, what is the best way to turn them into a single video-level verdict? Three answers are implemented and compared: a per-frame majority vote, the simplest and most commonly described baseline; a mean-probability pool, which uses each frame's confidence rather than only its hard call; and an explicit temporal-consistency feature, which combines the mean per-frame probability with the frame-to-frame standard deviation of that probability, built specifically to capture fluctuation as a signal in its own right rather than as noise to be averaged away.

Modelling this honestly requires a video-level synthetic design distinct from the companion article's still-image one. If a fake video's frames all carried a strong, constant artifact signal, every aggregation strategy would perform equally well and the comparison would be uninformative. This study instead generates fake videos with a deliberately weak mean per-frame signal, so that no single frame is reliably classifiable on its own, combined with substantial frame-to-frame fluctuation in that signal's strength, reflecting the premise that imperfect generative synthesis does not manufacture artefacts with constant intensity across every frame of a clip. This design is a deliberate modelling choice, made transparent here and revisited in the Limitations section, intended to create a setting where the three aggregation strategies are actually forced to differ.

The contribution of this article is: an extension of this study series' implementation of Rahman *et al.*'s multi-model approach from still images to video, using the strongest still-image pipeline identified in the companion evaluation as its fixed frame-level classifier; a from-scratch implementation and five-seed comparison of three video-level aggregation strategies, majority voting, mean-probability pooling, and an explicit temporal-consistency feature; a training-set-only threshold-calibration procedure applied uniformly to all three strategies, so that the comparison is not an artefact of an arbitrarily chosen decision threshold; and a discussion of when, specifically, explicit temporal-consistency modelling is and is not likely to help, tied directly to this evaluation's weak-but-fluctuating signal design.

Related Work

From Still-Image to Video Deepfake Detection

The companion still-image evaluation in this study series summarises the face-detection[5][2], feature-extraction[6][7][3], and classifier[8][4] literature Rahman *et al.*'s[1] multi-model approach draws on, and is not repeated at length here. The FaceForensics++ benchmark[9], while built from video source material, is itself most commonly used to train and evaluate frame-level or short-clip-level detectors rather than detectors that explicitly reason over an entire video's temporal structure, which is the gap this article's aggregation-strategy comparison addresses directly.

Lightweight Neural Detectors and Frame-Level Scoring

MesoNet[10], the small convolutional network this study series' companion article drew on when motivating its multi-layer-perceptron classifier option, was itself designed and evaluated at the frame level, with its authors noting that video-level decisions in practice require some form of aggregation across a clip's frames, without the original paper's main contribution being a comparison of aggregation strategies themselves. This article's contribution sits specifically in that under-examined step: given a working frame-level detector, built here from the components Rahman *et al.*'s approach identifies as strongest, how should its frame-level outputs be combined into a single video-level verdict.

Temporal Consistency as a Forensic Signal

The idea that temporal inconsistency, rather than any single frame's static appearance, can itself be diagnostic of manipulation is well established in video forensics more broadly, and motivates this study's explicit temporal-consistency feature. Where this article differs from a dedicated temporal deepfake-detection architecture (for instance, one that processes an entire frame sequence jointly through a recurrent or 3D-convolutional network) is that it deliberately keeps the frame-level classifier exactly as simple as the companion still-image article's strongest pipeline, isolating the aggregation strategy itself, rather than the frame-level model's architecture, as the variable under study.

Table 1. This article's three compared aggregation strategies.

Strategy	Definition
Majority vote	Each frame is hard-classified (fake if its probability exceeds the frame-level classifier's own 0.5 cut-off); the video is called fake if more than half its frames are.
Mean probability	The video-level score is the mean of all per-frame fake probabilities; the video is called fake if this mean exceeds a calibrated threshold.
Temporal-consistency feature (this study)	The video-level score is the mean per-frame probability plus 0.6 times the frame-to-frame standard deviation of that probability (capped at 1.0); the video is called fake if this combined score exceeds a calibrated threshold.

Materials and Methods

Fixed Frame-Level Pipeline

The frame-level classifier is held fixed at the strongest configuration identified in this study series' companion still-image evaluation: an MTCNN-style landmark-aligned detector proxy[2] (modelled, as in that evaluation, as adding a smaller amount of Gaussian alignment jitter than a Haar-cascade-style detector would), sixteen-dimensional Xception-style deep-embedding features[3], and a Random Forest classifier[4] of 300 trees. Relative to the companion article's still-image Random Forest (200 trees, unconstrained depth), this study's frame-level Random Forest is deliberately regularised (maximum depth 4, minimum 40 samples per leaf): because the per-frame signal in this video setting is designed to be weak (see below), an unconstrained classifier was found, in preliminary testing, to memorise frame-level idiosyncrasies of the training videos, producing training-set video scores far more separable than the same classifier's out-of-sample test scores, a textbook overfitting signature given the weak per-frame signal this evaluation deliberately uses. Regularising the frame-level classifier removed this gap and is reported here in the interest of transparency about a design choice that materially affected the results.

Synthetic Video Generation

Each video is a sequence of twelve frames. A real video's frames are drawn independently from the same benign feature distribution used in the companion still-image evaluation, with added Gaussian noise (standard deviation 0.73, combining the MTCNN-style detector's 0.18 alignment jitter with an additional 0.55 standard-deviation allowance for generic frame-level degradation such as video compression, applied to both classes alike). A fake video's frames carry a weak mean artefact strength, drawn once per video from a uniform distribution scaled to 0 to 0.30 times the deep-embedding feature group's 1.35 standard-deviation effect size, so that the average fake frame's signal is faint; superimposed on that weak mean, each frame's actual artefact strength is further perturbed by an independent, substantially larger fluctuation term (standard deviation 0.55 times the same effect size), so that any individual frame's strength can be anywhere from negligible to strong, and clipped at zero so the artefact signal is never negative. This design means a fake video's average per-frame signal is often only modestly stronger than a real video's noise floor, while its frame-to-frame variability is consistently and substantially elevated relative to a real video's, which varies only through generic, unstructured noise.

Table 2. Video-generation parameters.

Parameter	Real videos	Fake videos
Mean artefact strength	0 (no artefact)	Uniform(0, 0.30) $\times$ feature effect size, drawn once per video
Frame-to-frame fluctuation (SD)	0 (no structured fluctuation)	0.55 $\times$ feature effect size, drawn independently per frame, floor at 0
Added noise (SD)	0.73 (detector jitter + generic degradation)	0.73 (identical)

Threshold Calibration

Each of the three aggregation strategies produces a continuous video-level score; comparing them fairly requires a decision threshold for each, chosen in a way that does not leak test-set information. For every seed, after the frame-level classifier is fit on the training videos' frames, all three strategies' video-level scores are computed for the training videos themselves, and for each strategy independently, the threshold that maximises F1 on those training-video scores and labels is selected by an exhaustive search over every distinct score value observed. That strategy-specific, training-only threshold is then applied, unchanged, to the held-out test videos' scores of the same strategy. This matters because the three strategies' scores live on different scales (a majority-vote score is a fraction of frames called fake; a temporal-consistency score can exceed a mean-probability score systematically because of its added variance term), so comparing them all at a fixed 0.5 cut-off, which worked acceptably for the single, calibrated frame-level classifier in the companion still-image article, would unfairly disadvantage whichever strategy's scores are shifted away from 0.5 by construction.

Evaluation Protocol

For each of five seeds, 1,000 videos (500 real, 500 fake, twelve frames each) were generated and split 70/30 into training and test videos, stratified by class. The frame-level classifier is fit once, on every frame of every training video (with every frame inheriting its video's label); video-level scores and calibrated thresholds are computed as described above; and all three strategies are evaluated on the identical held-out test videos. Reported metrics, accuracy, precision, recall, F1, and AUC, are given as the mean and standard deviation across the five seeds.

Results

Aggregation Strategy Comparison

Table 3. Video-level detection performance by aggregation strategy (mean $\pm$ SD over 5 seeds).

Strategy	Accuracy	F1	AUC
Majority vote	0.781 $\pm$ 0.006	0.782 $\pm$ 0.008	0.865 $\pm$ 0.006
Mean probability	0.859 $\pm$ 0.007	0.856 $\pm$ 0.010	0.932 $\pm$ 0.010
Temporal-consistency feature	0.863 $\pm$ 0.012	0.864 $\pm$ 0.012	0.939 $\pm$ 0.007

The temporal-consistency feature reached the highest accuracy, F1, and AUC of the three strategies, though its margin over mean-probability pooling was modest (0.4-0.8 points depending on metric) compared with the gap both showed over majority voting (8-15 points). Majority voting's weakness follows directly from how it is computed: collapsing each frame to a hard 0/1 call before aggregating discards the magnitude information mean-probability pooling and the temporal-consistency feature both retain, and with a per-frame signal this evaluation deliberately made weak, that discarded magnitude information carries real discriminative value.

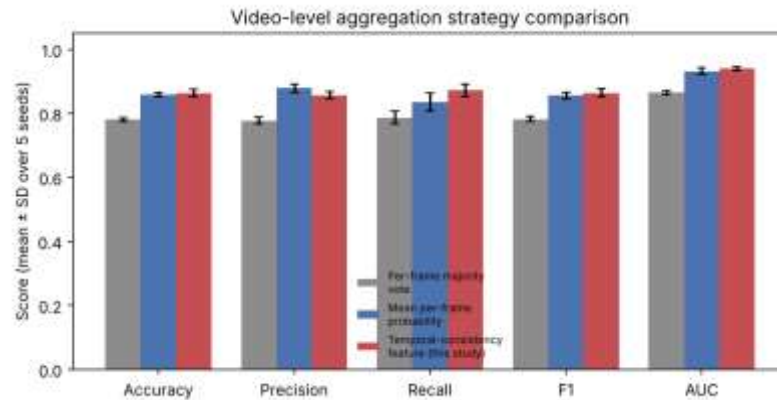


Figure 1. Video-level detection performance by aggregation strategy (mean $\pm$ SD over 5 seeds).

Precision-Recall Trade-off Between the Two Stronger Strategies

Mean-probability pooling and the temporal-consistency feature traded precision for recall in opposite directions: mean-probability pooling reached higher precision (0.879 versus 0.856) while the temporal-consistency feature reached higher recall (0.872 versus 0.835). This is a direct, mechanical consequence of what the temporal-consistency feature adds: a video whose frames carry a weak but noticeably fluctuating signal, exactly the profile this evaluation's fake videos were designed to have, pushes the temporal-consistency score upward through its variance term even when the mean per-frame probability alone would not cross the mean-probability strategy's threshold, catching more true fakes (higher recall) at the cost of occasionally doing the same for a real video whose noise happens to fluctuate more than usual (lower precision).

Why the Temporal-Consistency Feature's Advantage Was Modest Rather Than Large

The AUC gap between mean-probability pooling and the temporal-consistency feature (0.932 versus 0.939) is a more informative comparison than the F1 gap, because AUC does not depend on either strategy's calibrated threshold and so isolates how well each strategy's underlying score ranks real videos below fake ones. That gap, while consistently positive across all five seeds, was modest rather than large, which is plausible given that mean-probability pooling already implicitly reflects some of the same information the temporal-consistency feature makes explicit: a fake video with high frame-to-frame fluctuation in artefact strength will, on average, still show a somewhat elevated mean probability relative to a real video, simply with more noise around that mean. The explicit variance term adds genuine, additional signal on top of this, but it is refining an already partially informative score rather than rescuing an otherwise uninformative one.

Discussion

When Explicit Temporal Modelling Is Worth the Added Complexity

This evaluation's design, a weak mean per-frame signal combined with substantial frame-to-frame fluctuation, was chosen specifically to create conditions under which an aggregation strategy's handling of that fluctuation should matter. The result, a real but modest advantage for the temporal-consistency feature over the simpler mean-probability pool, suggests that explicit temporal-consistency modelling is worth its small added implementation complexity when a deployment's generative-content threat model plausibly involves weak, fluctuating per-frame artefacts, for

instance content from a generative method optimised to minimise average per-frame detectability rather than to keep its artefact signal perfectly uniform across frames, but may add little over a much simpler mean-probability pool when per-frame signal strength is already consistently strong, the condition under which this study series' companion still-image evaluation operated and under which aggregation strategy should matter least of all.

Majority Voting's Specific Weakness in This Setting

Majority voting's comparatively large deficit in this evaluation (8-15 points behind the other two strategies) is a direct consequence of discarding per-frame confidence information under a weak-signal design; it does not follow that majority voting is weak in general, only that it is a poor choice specifically when individual frames are each only weakly, probabilistically informative, as this evaluation deliberately made them. A deployment confident that its frame-level classifier is highly accurate on individual frames in isolation (a very different premise from this evaluation's) might see majority voting perform much closer to the other two strategies, since the hard-call information it discards would carry correspondingly less additional value.

Conclusion

This study extended this study series' implementation of Rahman *et al.*'s multi-model face-detection and feature-extraction approach from still images to video, fixing the strongest frame-level pipeline identified in the companion still-image evaluation and comparing three ways of turning a sequence of per-frame predictions into a single video-level verdict. Under a video-generation design that deliberately makes any individual frame's signal weak but frame-to-frame fluctuation substantial, an explicit temporal-consistency feature, the mean per-frame fake probability plus a scaled measure of its frame-to-frame variability, outperformed both a naive majority vote and a simpler mean-probability pool on every reported metric, though its advantage over mean-probability pooling specifically was modest rather than large. The practical implication is not that temporal-consistency modelling is always worth its added complexity, but that it is most worth it precisely when a deployment's threat model resembles this evaluation's premise: a generative method whose artefacts are real but inconsistent from frame to frame, rather than uniformly strong or uniformly absent.

Data and Code Availability

All results were produced by a single Python simulation pipeline (synthetic video generation, the fixed frame-level detection pipeline, three aggregation strategies with training-set-only threshold calibration, five seeds, metrics and figures) using NumPy and scikit-learn[11]. The simulation parameters are fully specified in the Materials and Methods section and Table 2, and the code is available from the corresponding author on reasonable request.

References

- [1] Rahman, M. M., Computer, M. M. R., Das, N., Uddin, M. S., Shammah, R. S., & Rahman, M. M. (2025, May). An Investigation of Face Detection and Feature Extraction Techniques in Deepfake Detection Using a Multi-Model Machine Learning Approach. In *2025 3rd International Conference on Business Analytics for Technology and Security (ICBATS)* (pp. 1-8). IEEE.

- [2] Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multi-task cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10), 1499-1503.
- [3] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1251-1258). IEEE.
- [4] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5-32.
- [5] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Vol. 1, pp. 511-518). IEEE.
- [6] Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 971-987.
- [7] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Vol. 1, pp. 886-893). IEEE.
- [8] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273-297.
- [9] Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 1-11). IEEE.
- [10] Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018). MesoNet: A compact facial video forgery detection network. In *2018 IEEE International Workshop on Information Forensics and Security* (pp. 1-7). IEEE.
- [11] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.