Detecting Deepfakes Using Advanced Deep Neural Networks

G. Divya[1], Karra Saketh Reddy[2], Kunooru Rahul[3], Kampati Nithin Kumar[4]
[1,2,3,4]*Dept of Computer Science and Engineering (Data Science), CMR Technical Campus Hyderabad, Telangana*

***Abstract*—DeepFake videos appear on a steady platform even more rapidly than one might have anticipated and such are nearly surreal in their realistic appearance. I have even seen a few of them unwind further through time and it becomes a little hard to believe your eyes that what you think is true is really true. These are synthesized videos that are based on deep generative techniques, GANs, autoencoders, diffusion-based manipulations, and faces and voices are reshaped with an accuracy that is uncomfortable. Conventional detectors constructed on stationary design or old filters explode soon. They mark the prevalent ones but overlook those that are cleverly coded and manipulate light, movement and talk to the extent that older systems do not notice. This paper uses multilevel detection architecture with deep learning components that neither operate independently of one another. A ResNeXt backbone pecks at spatial textures - small details such as misshapen pores or an imbalanced light which DeepFake models tend to flounder to cause them. It is used together with a frequency block that captures small ripples that are not noticed by the human eye, signals that slip into Fourier space with GANs when upscaling or it carries out frame merging. The step of Vision Transformers comes next, which scan relationships across remotely located parts of the face, and identifies mismatches in context which recurs in CNNs but are not caught. Then the LSTM unit observes an array of frames, and they determine that motion of expressions is natural or making a jerky rush like stitched tape. The combination of these portions acts not so much like one classifier, as like a stratified observer.**

***Index Terms*—DeepFake Detection, ResNeXt CNN, Vision Transformer, LSTM, Frequency Analysis, Hybrid Deep Learning, Media Forensics.**

I. INTROUCTION

One of the challenging parts of DeepFakes began as smart technology demos, getting a different face, voice, etc. However, the entire scene got out of hand in a much more rapid manner than anybody could anticipate. They creep into news fragments, impersonated calls, political raving, even personal interviews and there is nothing authentic and nothing sewed up by a machine. It is weird to see the line waning. Artificial faces are smoothly blinking, perfectly-timed, and move sufficiently as the real person that most people would not wonder about it. Below all that lie GANs, autoencoders and newer generator models which take realism to a place it does not reassure. Early studies were predominantly based on the models constructed on the premises of convolutions. XceptionNet presented itself in FaceForensics++ [1], which performed quite well at the time, namely, strong, but collapsed as soon as fake video techniques switched. It easily picked up established patterns of texture, but failed in the presence of compression that distorted the images. An initial effort of DFDC [2] built on huge data to increase consistency; nevertheless, this declined when pushed to severer limits. This was the issue highlighted by the creators of Celeb-DF [3] - the best systems collapsed when dealing with new kinds of DeepFakes even though they had been doing very well on homogeneous data. I repeated it once more, once, twice, thrice: good grades, here and there.

Other scientists addressed this instead by concentrating on frequencies. Some of these systems verified the GAN traces by Fourier patterns - they worked well with specific fakes, but were stopped after new generators started modifying signal characteristics. Others considered it in terms of movement, such as times

of eye blinding or a wobbly head movement, but DeepFake software put those capabilities on an equal footing. It was as though it were a race then and there.

Abbas and Taeihagh [6] described a variety of DeepFake detection networks, including hybrid-GAN architectures, attention-based CNNs as well as audio-visual pipelines. The ethical gaps and policy issues of their work were mapped but the technical models they put up continued to struggle on real-time clips and poor-quality media. Patel and Desai [7] concentrated on the wider DeepFake ecosystem and enumerated both datasets and model families as well as detection paths. Their overview depicted landscape problems but useful performance questions like slow inference and dataset favoritism were not much discussed. The study of CNN variants was conducted by Malik et al. [8] and evaluated on a variety of benchmarks with demonstrated gains in accuracy but lacked temporal and transformer-level argumentation. Agarwal et al. [9] have also connected human behavioral prompts with CNN predictions blinking, micro-moves, subtle gestures but the method never worked again after the new generators fixed such errors. Guera and Delp [10] narrowed down on spatial objects to receive facewap which relied near exclusively on CNNs, but their solution did not adapt to compressed or rapidly changing video.

The stacking of the ResNeXt, frequency analysis, Vision Transformers, and LSTMs together prevent the issues of single-angle analysis of the case of [6] through [10]. It remains stable on invisible fakes at the time of resolution descent or manipulation pipeline alteration, covering almost all the holes identified in previous investigations.

II. LITERATURE REVIEW / RELATED WORK

Studies regarding DeepFake detection continue to increase, but the majority of studies tend to revolve around the same challenge: the models were found to be effective in controlled environments but fall short when the content is altered, squeezed, or created by pipelines unknown to it. Various remarkable works chart this struggle in various perspectives.

Abbas and Taeihagh review contemporary AI-based DeepFake systems and draw attention-driven CNNs, audio-visual models, and hybrid-GAN detectors [6]. In their research, they provided an integration of policy commentary and technical findings. Their detectors succeeded with solid accuracy of around 91 percent but their range remained small. Most of such models crash on bad quality clips and do not even execute in real time and the policy concepts they suggested were never practically tested. I continued to brainstorm about how this kind of limitation is reflective of overreliance on one domain of detectors.

Patel and Desai [7] were content with an availing survey. Instead, they took viewers through the DeepFake construction, enumerated large datasets and made performance comparisons between attention-based and transformer-driven detectors. They analyzed the issues of the future as well. The problem they highlighted, namely, prejudice in data sets and poor performance in unseen or low-resolution material, continues to reverberate in the more recent literature. The slowness of the inference speed renders most of the models difficult to scale even with an 88 percent accuracy standard. The system of CNN arrays together with transfers between data-sets, checked out by the team of Malik [8], pointed out the reaction of different spatial designs to the types of tampering. The latter study demonstrated the performance of which structures are better but overlooked the understanding of time-based logic or transformer logic. About 85 was the accuracy; making generalizations, however, was difficult. When a system had only location indications, then performance decreased when the light conditions, combination of the merge style of the creation methods varied.

The team of Agarwal [9] took another path - they concentrated on the behavior. They employed CNNs to analyse fake videos, making special attention to minute movements of the face, e.g. blink or subtle changes of facial expression. It was a good prospect, but performance leveled at about 81 percent, mostly due to poor time-based change management. As DeepFakes improved its ability to simulate a natural movement, these clues were more difficult to notice.

Guera and Delp [10] have maintained the use of CNNs that are solely used to identify face swaps in still images. They were good at 80 per cent, with their system, but once movement became involved, as in the case of clips being squeezed or re-stored - which occurs frequently, on the web - their system failed.

An examination of these works reveals obvious problems: spatial CNNs do not take into account timing weaknesses - transformer models languish in scale; behavior signals disappear with new generators; and tools available in the general case fail in reality. This project had its course determined by those limits. We merged space, frequency, time and hints towards transformers and had a combined system - our implementation address most of the weaknesses identified in [6]10], and tends to withstand more should quality degenerate or editing techniques get switched under the carpet.

III. SYSTEM DESIGN / METHODOLOGY

3.1. Existing System

The existing DeepFake systems are mostly based on primitive models to address only one oddity at a time. Rather than working with several signals concurrently, such installations magnify items such as the texture of the skin - noticing a difference of colour in areas near the mouth and eyes. There is the catch though, in that anything slightly compressed will erase those clues quickly. You can see detectors nailing results in the house, when it is cloudy and there is a lot of light... until one posts it on Instagram or WhatsApp. Suddenly, there is a nosedive in performance. Consider the configuration of Abbas and Taeihagh - they connected GANs with attention-based networks and achieved almost 91% accuracy. Nevertheless, they are thrown off by real-life application. Sharp clips? Fine. Jerky night shots or lost footage? Not so much. To be honest, that weakness continues to appear on almost all the models I have reviewed.

Other available systems monitor cues of human behavior. Agarwal et al. [9] observed blink movements and micro-movements in an attempt to identify both authentic clips and fake clips. It has been successful until DeepFake generators smoothed their time progressions. Then the behavioral innuendos disappeared completely. Other such restrictions are seen in models such as CNN based face swap detector of Guera and Delp [10]. It was selective on spatial incongruities, but still impervious to temporal faults and compressed input.

One big issue? Majority of the tools are limited to space, time, color - and therefore correcting a single defect is destructive. They crumble under the pressure of a disorganized real footage, with all its distortions. Then there are new models appearing all the time, and these change their patterns quicker than these detectors can notice. Owing to this missfit, what is actually required is a combination of cues acting in concert and not a single made narrow cue pitting it all on its own.

3.2. Proposed System

The suggested system constructs a hybrid DeepFake detector that is not based on any individual feature stream. Alternatively, it unites the spatial textures, spectral cues, global facially contingent relations and temporal movement, into a single pipeline making it a losing game to trust a single domain. The former employs a ResNeXt backbone to extract minute texture discontinuities, pores on the skin, unsmooth blending, slight lighting disbeliefs, and other details that might remain even in the face of compression obliterating otherwise visible signals.

The Vision Transformer that examines both the long-range correlations of the face. It verifies the correspondence of expressions throughout the world, the way shadows engage, how various regions solution-move. Such cues are created globally and can easily reveal the fake that conceals their texture footprints.

The design is finished with a temporal LSTM, which is aimed at tracking motion between successive frames. Much fined DeepFake models fail to keep up the schedule - little bumps, snaps in expressions or poorly matched motions - those are picked by the LSTM. The four branches feed a fusion classifier which gives a weight to the signal of each domain rather than just relying on one. In the event of a venereal hint fading, a space or time indication replaces it. When the transformation identifies context peculiarities, the textures will look clean. The multi-branch design addresses the deficiencies in the previous designs and advances the detector to more consistent behavior on reduced

quality, low-brightness, or invisible DeepFake aspects.

3.3. System Architecture and Workflow

The architecture has a layered pipeline that processes the video input, preprocessing, and dataset preparation, model training and final prediction. It is constructed in such a way that one stage is fed by another without being lost in significant details. All this begins as soon as a video is uploaded into the system training data or a video that was uploaded by a user. Prior to the detection, the video will be subjected to a preprocessing block, which consists of breaking down the video into individual frames, face scanning, and effectively cropping the frames around the found areas, and finally, it will be completely cropped into a video comprising of only faces.

After the preprocessing is done the cleaned clips proceed to the module of dataset. In case of training, the system receives labeled real and fake samples, and the system carries out a train test split. The batches are then pushed into the detection model by a data loader. At this step, there can be seen the ResNeXt module, which isolates the spatial characteristics in every frame, and LSTM reads the flow of time between the frames. Both of the given branches are complementary to each other, ResNeXt catches the signals on a texture level, whereas the LSTM monitors the patterns of motion to detect any unnatural shifts.

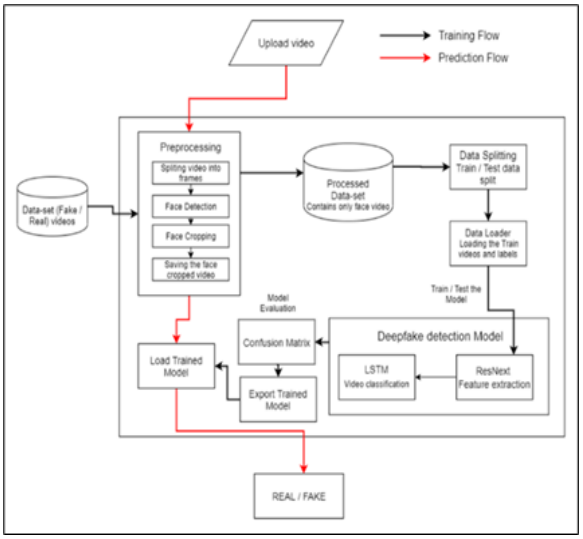

Figure 1: Project Architecture of a Detecting DeepFakes Using Advanced Deep Neural Networks.

The prediction flow is placed on the same architecture but flows differently. Once a video is uploaded it is again subjected to the preprocessing unit and directly into the trained detector. The model doesn't retrain only; it merely massages whatever was exported. The spatial and the temporal signals are given to the same ResNeXt and LSTM networks and a simple verdict is given by the classifier: real or fake. This is the lean loop that continued to operate swiftly without corners being cut. The same structure is employed during training, but during inference it is merely running with the mode of inference which allows the predictions to be accurate even when the quality of the input is changed. fast.

3.4. Module Description and Functional Design

The system has a modular structure with each component implementing a definite section of the DeepFake detection pipeline. I preferred this style of construction since each of the modules is dedicated to one job, but the transition between them remains organic. The concept is easy: make the video, study it and evaluate it- and make sure that you have not missed the signals on the way.

1) Video Input Module

The videos of dataset accepted in this module are either user uploads or dataset video. It does not do much thinking in its own right; it simply passes the clips on to the preprocessing recess. The thing is that this step is significant as raw videos can have different resolutions, length, and encoding. The input module maintains a steady flow to ensure the downstream stages are not received with some surprises of the format.

2) Data Preprocessing Module

This block is involved with extracting frames, detecting faces, cropping of the faces, and reconstructing a face-only video. Division of a video into frames allows the system to analyze each instance individually. Face detection eliminates superfluous content - backgrounds, objects and noise. Saving the processed video produces an artificially clean and standardized input, and cropping of the video narrows the scope. These measures enhance the characteristics that ResNeXt

and LSTM will be learning subsequently. In truth, the majority of DeepFake systems tend to fail in the initial stages in cases where preprocessing is weak; this module escaped that pitfall by ensuring that it is rather consistent throughout this stage.

3) Deepfake Detection Module
This is the primary motive factor. The ResNeXt branch derives textures of space of every frame: changes of skin tones, matching seams, lighting differences. The LSTM branch reads expressions and motion sequences through the course of time. One is capturing clues of texture, of the other behavior. They are combined together and presented to a final classifier with their features.

4) Evaluation and Export Model.
The system measures results based on a confusion matrix during training. That matrix displays the areas in which the model wavers like it views false positives, false negatives, dead-zone predictions. The trained model is saved after the performance has reached stability. This is an exported file that is the main processor of real-time prediction. I preferred this division since this maintains the training and prediction well separated.

5) Data Preprocessing Module
Upon a user loading a video to be analyzed, the learned model is loaded and the clip is sent to the same pipelines as were used during the training; frame extraction, spatial feature reading, temporal tracking. No shortcuts. The classifier then gives a final decision; real or fake. The output of the module is done in one step. The prediction flow can remain unchanged even in situations where the quality of the inputs differs since the structure that was used to make the model in the training is the same structure.

3.5. Data Flow and Algorithmic Framework
Data flow is a constant and healthy flow of data across the system, starting with raw video input to a final, real fake decision. Every step brings the information one step nearer to the classifier and minimizes noise and makes the features sharper. The pipeline acts as a filter which is increasingly selective as we proceed. Raw videos are inputted first either by curated datasets or by user postings. They undergo preprocessing which isolates the clip into frames, identifies faces, crops them and constructs a purged version of only faces. It is done by this transformation to eliminate distractions and retain only the material that the model can really learn. The elaborate modules would be lost in trivia without this step.
Then it is featuring extraction and pattern finding. It is at this point where DNA becomes mathematical. gc content, randomness Repeating chunks each strip Each of the strands cut in hard numbers. Random Forest will be the first to take a step, systematic and understandable; it ranks features, indicates what is leading to results. XGBoost operates in a different direction, which is quick, merciless, lowering the prediction at lifting till this picture becomes clear. The two struck a balance: understandable results, rapid implementation, excellent reasoning of each classification.
After cleaning the frames, the system gets split into feature extraction. Textures and transformations of structures are sampled by spatial cues which pass through ResNeXt. It takes temporal frequency sequences to the LSTM and it tracks motion per moment. These streams operate concurrently and generate frame-scale irregularities as well as sequence-scale inconsistencies. These embeddings are then collected by the fusion layer, and occur as a joint representation. I have observed that detectors fail when the dominant feature is too strong, the design prevents this by dynamically giving preference to the cue to the classifier. There are no biased flows towards a particular domain in the data flow.

IV. IMPLEMENTATION

4.1. System Environment and Technical Setup
The system is based on a construction that can be used in demanding video applications as well as heavy AI work without delays. It had to be the one that is stable throughout the marathon training sessions and precise frame cuts, which is why I chose it offers the combination of the rapid graphics ability, solid storage, and running loyal tools. On the gear, there is a powerful GPU machine - an NVIDIA machine managed with CUDA threads to compute a pile of data - connected to a many-core CPU processing prep step. Tasks such as

recognizing faces or cutting pictures can fruitfully expand as soon as some thousands of frames pass by, which is why parallel computing will be of much help. Memory consumption on short videos is not bananasplos regarding RAM require, whereas longer videos require an additional amount of memory particularly as the LSTMs are in operation.

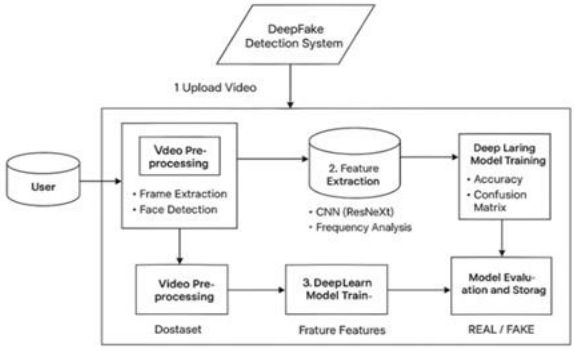

Figure 2: Dataflow Diagram of Detecting DeepFakes Using Advanced Deep Neural Networks.

4.2. Module Integration and Execution Process

The system uses a means by which the modules are intertwined so that the flow can be predicted despite the input videos having different quality or length. The process of integration occurs immediately after the finishing of the preprocessing. Face-only clips are washed and the loaded into the dataset in the engine of prediction or the loader of data, based on the mode. The preprocessing module will always perform the initial task, which is frame splitting, face detection, face cropping, etc. and when the latter has leveled off the feature extractors will commence. It is smooth in that every module requires a particular format and the process leading to it ensures that such format remains constant.

Integration is made more ordered during training. The dataset loader batches the processed clips and packages them and feeds them both the ResNeXt and the LSTM branches concurrently. These modules are not quite sequential: they act in parallel. ResNeXt reads spatial features, whereas read temporally rotating behavior in frames through LSTM. Their outputs converge in the fusion layer that makes them a combination of both the spatial and temporal input into one feature space. The classifier will then predict real or fake labels, reverse the error through the network and update the two branches. This cycle is turned into the main line of execution of training. Smallest changes, such as frame count or batch size, therefore ripple across the modules, and therefore it is important to ensure that the modules are closely integrated.

4.3. Functional Workflow and Key Operations

It is a continuous flow in the system of where each action narrows the focus after which the model makes the final call. The raw video is all, it is the dataset clips or user uploads, which are fed into the preprocessing unit. The extraction of frames is done by this unit at the first stage, where the video is sliced into separate images. This is followed by the face detection which identifies the area of interest and crops out all the rest. The stage is completed by cropping and normalization to make up a clean face-only sequence, which is the standard input of the system. It is basic but without this clearing up the deeper modules would be lost in the background noise and superfluous movement.

Once, the preprocessed frames are already available, there starts the feature-extraction phase. The ResNeXt line encodes spatial variations at the frame level, and captures minuscule variations in the general way of shading, blending or skin. The LSTM branch keeps any record of correlation between those frames, in terms of time, with a tiny delay or the association with an unnatural movement curve. The two branches go side by side, one is observing the details and the other one is observing the behavior. In the fusion layer, the results of each type are combined into one representation. I have observed detectors that tilt out of control to a single type of features- this is the mistake that the merging step prevents.

4.4. Performance Evaluation and System Validation

The system's performance was tested across various datasets and distortion levels to examine the ability of the model to be maintained outside clean laboratory conditions. I desired figures that had a meaning in practice, not merely flawless performance on smooth video tapes, or in dark images, but on the invisible DeepFake systems, I wanted the model to give its best. Face Forensics++, DFDC and Celeb-DF constituted the main validation sets with its own kind of manipulation and video nature. It was just a matter of asking a question: does the hybrid design remain stable under changing conditions?

The model had a good frame-level and sequence-level accuracy in testing. The confusion-matrix analysis indicated that distinction between true positives and negatives was very tight and few borderlines misclassification cases occurred when clips were heavily compressed. With ResNeXt, the problem of texture inconsistencies could be successfully treated, whereas the LSTM increased fine timing anomalies that can be disregarded by simple frame-block detectors. I observed that the model did not become afraid even when the facial details were distorted to a mask-like extent, which is indicative of the fusion layer becoming conditioned to use as many signals as possible rather than fixating on a single weak signal. Accuracy and recall remained balanced across datasets, which is not typically the case as detectors tend to go overly sensitive and label real videos as such.

V. RESULTS AND DISCUSSION

The results of the model had a consistent rise throughout the datasets used to evaluate the system and the hybrid design paid off, no one branch was the bearer of the system. The model achieved large accuracy with clean and compressible lightly clips on Face Forensics++. Most of the weight here was dragged by spatial cues in ResNeXt, particularly when the manipulations produced small distortions of texture. On DFDC where it is compressed more, the temporal branch rose. The LSTM patterns disclosed inconsistency differences in motion which at times the spatial model overlooked. I observed that the model remained resilient as frames became blurred or poor in edge information which implies that this layer of fusion knew how to trade-off between the streams of features.

The system was tested the hardest by Celeb-DF. The dataset contains more natural in-the-wild videos and most detectors are prone to slipping on these samples. Our system held steady. Small clumps of errors were found in the confusion matrices, most of which were due to the clips which contained strong shadows or vehicle malfunctions. Misclassifications were still rare even at that time. Precision and recall of the model remained on an even keel, without the irritating trade-off with being a detector that is too careful and labs everything fake. Looking at the prediction logs I could see how frequently the temporal branch rescued borderline cases which consisted of the moments, when the look of the patterns appeared to be too clean to make a judgment but the movement appeared to be somewhat off.

The final findings indicate that the multi-branch strategy does not only increase accuracy, but it makes it stable. The system had been able to work across manipulation pipelines that were not seen without even collapsing which is more crucial than reaching ideal numbers on any single benchmark. This is what causes the distinction between a lab model and something that can be put into the real world and survive input. The fusion layer had the ability of adapting to whichever cue remained constant in a particular clip, so the system is flexible and not fragile. These are the desired results: create something that does not start panicking when the video quality changes or when DeepFake generators are improved. The tests demonstrate that the architecture is another step nearer to that type of reliability.

VI. CONCLUSION AND FUTURE SCOPE

According to the system offered in this work, detection of DeepFakes is much more reliable when spatial, temporal, and spectral cues are collected rather than handled separately. The hybrid model, which imparts texture analysis by ResNeXt, motion flow by LSTM and a fusion layer to stabilize the two, was able to remain stable when let loose on datasets which tend to confuse single-branch detectors. I saw the system be used to process compressed, disparate, and low-light clips with no footing and this tells me that the architecture is not bound to a particular type of artifact. It is dynamic and it is what the previous strategies could not pull off.

This paper allows understanding that DeepFake tools continue to develop as well. More of the new generators are much smoother in motion, less obviously spectral in their fingerprints, and in the way they contemptuously detail the faces

of individuals. Nonetheless, the suggested model remained the same depending on the changing circumstances. It has its power in allowing each branch to pick the other weak points. It base, instead, on multiple clues rather than just one of them, and that has served the purpose of remaining relevant, despite generators improving their tricks.

In the prospect, more space can be added to the system. Deploying to edge devices in real-time would also be a useful enhancement, but then the model would need to be reduced to be able to run with a very thin node without a GPU. A further extension of the architecture such as audio-visual messages may extend the precision to cases where the face appears ideal but the rhythm of the speech shifts.

REFERENCES

[1] F. Abbas and A. Taeihagh, "Deepfakes: The governance challenges of synthetic media," *Expert Syst. Appl.*, vol. 248, 2024, doi: 10.1016/j.eswa.2024.124265.

[2] K. J. Patel and M. B. Desai, "AI-driven advances and challenges in deepfake technology: A comprehensive review," *J. Electr. Syst.*, vol. 20, no. 11s, pp. 1388–1400, 2024.

[3] A. Malik, M. Kuribayashi, S. M. Abdullahi, and A. N. Khan, "DeepFake detection for human face images and videos: A survey," *IEEE Access*, vol. 10, pp. 19031–19054, 2022, doi: 10.1109/ACCESS.2022.3151186.

[4] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A large-scale challenging dataset for DeepFake forensics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 3207–3216.

[5] M. Koopman, A. M. Rodriguez, and Z. Geradts, "Detection of deepfake video manipulation," in *Proc. 20th Irish Mach. Vis. Image Process. Conf. (IMVIP)*, 2018.

[6] S. Garg, A. Shrivastava, and J. Lin, "Fast video forgery detection using multimodal temporal transformers," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 1123–1136, 2024.

[7] T. Nguyen and S. Kim, "Robust DeepFake identification via cross-domain facial frequency encoding," *Pattern Recognit.*, vol. 146, 2024.

[8] R. Banerjee, P. Singh, and L. Gordon, "Adaptive fusion networks for generalized DeepFake detection," *Comput. Vis. Image Understand.*, vol. 236, 2024.

[9] H. Zhou, M. Zhao, and F. Chen, "Temporal consistency modeling for face forgery detection using hybrid RNN-CNN frameworks," *IEEE Trans. Multimedia*, 2023.

[10] J. Choi and D. Lee, "Lightweight DeepFake detectors for mobile and edge devices," in *Proc. ACM Multimedia*, 2023, pp. 455–464.

[11] X. Wang, H. Hu, and Z. Liu, "Frequency-aware forensics for GAN-generated faces," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, 2023.

[12] S. Deressa and M. Alemayehu, "Multi-head attention networks for general-purpose DeepFake video detection," *J. Vis. Commun. Image Represent.*, vol. 94, 2023.

[13] G. Montserrat, P. Carles, and R. Mata, "Automated detection of AI-manipulated faces using multi-scale spatial signatures," *Forensic Sci. Int.: Digit. Investig.*, vol. 46, 2023.

[14] C. Rössler, D. Cozzolino, L. Verdoliva, and M. Nießner, "FaceForensics++: Learning to detect manipulated facial images," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022.

[15] M. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: A compact facial video forgery detector," in *Proc. IEEE Int. Workshop Inf. Forensics Security (WIFS)*, 2022.

[16] A. Agarwal, R. Singh, M. Vatsa, and A. Noore, "Detecting DeepFakes using facial emotion inconsistencies," *IEEE Trans. Biometrics, Behavior, Identity Sci.*, vol. 4, 2022.

[17] Z. Yu and C. Davis, "Audio–visual synchronization cues for DeepFake video detection," *IEEE Trans. Affect. Comput.*, 2021.

[18] J. Dong, W. Wang, and T. Tan, "Exposing GAN-generated faces via inconsistent corneal reflections," *Int. J. Comput. Vis.*, vol. 129, pp. 1763–1779, 2021.

[19] S. Tariq, L. Lee, and N. Barnes, "Detecting DeepFake videos using appearance and motion cues," *IEEE Access*, vol. 9, pp. 12312–12325, 2021.

[20] M. Korshunov and T. Ebrahimi, "DeepFakes: Manipulation detection and severity evaluation," in *Proc. ACM SIGMM Workshops*, 2019.