Global Scientific JOURNALS

GSJ: Volume 12, Issue 3, March 2024, Online: ISSN 2320-9186
www.globalscientificjournal.com

A Novel CNN-based Architecture for Automated Detection and Classification of Deepfake Images from Video

Dennis Lucky Tuanwi Bale
Department of Computer Science, University of Port Harcourt, Nigeria
Email: denlucky@gmail.com

Laud Charles Ochei, Chidiebere Ugwu
Department of Computer Science, University of Port Harcourt, Nigeria
Email: laud.ochei@uniport.edu.ng, chidiebere.ugwu@uniport.edu.ng

Abstract

Deepfake technology has become increasingly sophisticated, posing a significant challenge to the integrity of digital media and exacerbating the spread of misinformation. In response to this growing threat, researchers have explored various approaches to detect and classify deepfake images from videos. However, existing methods often suffer from limitations such as low detection accuracy, high false positive rates, and inadequate scalability. To address these challenges, this paper presents a novel CNN-based architecture for automated detection and classification of deepfake images extracted from videos. Our approach builds upon the advancements in convolutional neural networks (CNNs) and incorporates multimodal fusion techniques to enhance the model's ability to analyse visual, auditory, and metadata information simultaneously. The proposed architecture utilizes a combination of early and late fusion methods to integrate features from different modalities, thereby improving detection accuracy and robustness against adversarial attacks. Through rigorous scenario-based evaluation methods, we meticulously assess the performance of the proposed architectural framework across various operational scenarios. Our findings demonstrate the efficacy of the proposed approach in accurately identifying and classifying deepfake images from videos, even under challenging conditions. Furthermore, we identify the risk and trade-offs as well as the impact of changes on the architecture's performance.

1. Introduction

Deepfake technology has rapidly advanced, presenting significant challenges in maintaining the integrity of digital content and combating the spread of misinformation. The creation of fake images, audios and videos have recently become popular and easier because of the emergence of deepfake technology. Deepfake refers to manipulated digital media such as images of videos where the image or video of a person is replaced with another person's likeness (Almars, 2021). Deepfake is a term that comes from the combination of the concepts "deep learning" and "fake", referring to the fake content generated with deep learning which is Artificial Intelligence (AI) technology based. The content of the video is manipulated in such a way that people are made to say and do what they actually do not. The era of popular saying "seeing is believing" with respect to video content is actually being disproved by emergence of deepfakes. These give rise to propagation of fake news, fraud, identity theft, etc which are negative application of deepfakes. The need to control these negative trends of deepfakes application necessitate this research of automated deepfake detection and classification of images from video.

Motivated by this problem, automated detection and classification systems have emerged as crucial tools in identifying and mitigating the impact of deepfake content. Convolutional Neural Networks (CNNs) have shown promise in effectively detecting and classifying deepfake images and videos due to their ability to learn hierarchical representations of visual data. The aim of this research is to present a novel integrated CNN-based architecture automated detection and classification of deepfake images from Video. The specific contributions of this paper are:

1. Present a review of current literature on existing CNN-based architectures for detection and classification of deepfake images form videos.
2. Present a novel integrated CNN-based architecture automated detection and classification of deepfake images from Video but integrating multimodal fusion at the early stage and late stage of the CNN process.
3. Evaluation of the integrated CNN-based architecture automated detection and classification of deepfake images from Video
4. Presents the risks and trade-off in implementations of the architecture and the mpact of changes on different performance criteria for the architecture.

The current CNN-based architecture has been modified to include a novel multimodal fusion that happens in two stages. The first stage is the early stage of multimodal fusion, which happens after the input layer. The second stage is the late stage of multimodal fusion with metadata, which happens after the CNN process's convolutional layer. The evaluation of the CCN-based architecture based on the ATAM, a scenario-based method for software architectural evaluation reveals that this novel architecture can be used to detect and classify deepfake images from videos on social media platforms.

The rest of the paper is organised as follows: Section 2 is presents an overview of related concepts and related literature. Section 3 is discusses existing CNN-based architectures or detection and classicisation of deepfake from videos. Section 4 presents an evaluation of CNN-based architectures or detection and classicisation of deepfake from videos. Section 5 presents the results and including the discussion of the results related to the identification of risks and

trade-offs as well as the impact of changes on performance criteria. Section 6 concludes the paper with future work.

2. Overview of Related Concepts and Review of Related Literature

Several research have been carried out in the area of both deepfakes creation and detection. The availability of deep learning tools has made the creation of deepfake easier and affordable since it does not require sophisticated computing devices anymore. Generative deep learning algorithms have progressed to a point where it is difficult to tell the difference between what is real and what is fake (Mirsky & Lee, 2021).

2.1 Deepfake Creation

Deepfakes have become popular due to the quality of tampered videos and also the easy-to-use ability of their applications to a wide range of users with various computer skills from professional to novice. Deepfake techniques involve several deep learning algorithms to generate fake contents in the form of videos, images, texts or voices. Deepfakes are created using variations or combinations of generative networks (GAN) and encoder decoder networks (Mirsky and Wenke, 2020); (Mahmud and Sharmin, 2023).

To make a deepfake video, the developer first feeds countless hours of actual video footage to a deep neural network, which is then "trained" to recognize detailed rhythms and traits of a person. This is performed to provide the algorithm with a realistic representation of how that individual appears from various perspectives. The next thing is to combine the trained learning algorithm with computer graphics technologies to overlay real-time video of a person with AI-generated facial and vocal patterns obtained from neural network input.

The reduction of dimensions and compression of images in deepfake creation is achieved by the use of deep encoding model generated from deep autoencoders of deep neural network architecture with 4 or 5 layers representing encoding while the rest represent decoding (Cheng et al, 2019, Chorowski et al, 2019).

Face2face, a real time facial reenactment method proposed by (Fernandes *et al*, 2020) that works for any commodity webcam. Since this method uses only RGB data for source and target actor, it can manipulate real time Youtube video.

Notable deepfake creation in public domain especially on social media are the Barack Obama's deepfake by Jordan Peele and Tom Cruise's deepfake by Chris Umé. Both uses GAN and lip-syncing techniques to produce the deepfake videos.

2.2 Deepfake Detection and Classification

It is becoming increasingly difficult since synthetically generated faces are not only photo-realistic, they are almost indistinguishable from the real thing and are considered more reliable. Due to the negative application of the deepfake technology it more necessary to develop tools that can automatically detect and classify fake and real videos. These tools are to look out for the following traits of deepfake which are yet to be perfected by deepfake generative tools. The traits are Face and body reenactment, facial expressions and body movements or postures, video length usually short, video sound and lips movement especially

where lip-syncing is poorly done, and Misalignment of facial landmarks like mouth, nose and eyes.

Different methods have been proposed to detect the GAN generated images using deep networks. Tariq *et al* (2018) suggested neural network-based methods for detecting fake GAN videos. This method employs pre-processing techniques to analyses the statistical features of image and enhances the detection of fake face image created by humans (Li *et al*, 2018).

Raza *et al* (2022), proposed a novel deepfake predictor (DFP) approach based on a hybrid of VGG16 and convolutional neural network architecture. The deepfake dataset based on real and fake faces is utilized for building neural network techniques.

Hamza *et al* (2022), proposed the use of machine and deep learning-based approaches to identify deepfake audio. Rafique *et al* (2023), proposed an automated method to classify deep fake images by employing Deep Learning and Machine Learning based methodologies. Deepfake detection approach by Kaur *et al* (2022), uses the forged video to extract the frames at the first level followed by a deep depth-based convolutional long short-term memory model to identify the fake frames at the second level.

Durall et al (2020), proposed a novel approach to unmasking deepfakes using a method based on a classical frequency domain analysis (FDA) followed by a basic classifier. Hande et al (2022) propose a Novel Method of Deepfake Detection by using three different models CNN model, CNN-LSTM model, and CNN-GRU model to train and test on the DFD dataset, DFDC dataset, and Custom dataset, respectively.

Kosarkara *et al* (2023), developed a customized CNN algorithm to identify deepfake pictures from a video dataset and conducted a comparative analysis with two other methods to determine which one is superior. The Kaggle dataset was used to train & test the model.

2.3 Review of CNN-based Architectures for Automated Detection and Classification of Deepfake Images from Video

The rise of deepfake technology has necessitated the development of robust and efficient automated detection and classification systems to combat the spread of altered visual content. Convolutional Neural Networks (CNNs) have emerged as a popular solution to this problem due to their ability to learn hierarchical representations of visual data. This section provides a comprehensive review of existing CNN-based architectures used for automated detection and classification of deepfake images in video.

Early Approaches.

Early CNN-based deepfake detection architectures relied primarily on image-level features to identify inconsistencies and artefacts that indicated manipulation. FaceForensics (Rossler et al., 2019) and DeepFakeDetection (Li et al., 2020) used traditional CNN architectures such as AlexNet (Krizhevsky et al., 2012) and VGG (Simonyan & Zisserman, 2014), which were pre-trained on large-scale image datasets such as ImageNet (Deng et al., 2009). While these models performed reasonably well in detecting basic forms of manipulation, they struggled with more sophisticated deepfake techniques like facial reenactment and expression transfer.

Researchers have proposed specialised CNN architectures for deepfake detection and classification, addressing limitations in previous approaches. These architectures use novel components like attention mechanisms, temporal modelling, and multimodal fusion to improve detection accuracy and robustness.

FaceForensics++, for example, introduced a spatiotemporal CNN architecture that can detect both spatial and temporal dependencies in video sequences. By analysing motion patterns and temporal inconsistencies, the model achieved cutting-edge performance in detecting deepfake videos.

Deepfake detection relies heavily on transfer learning and domain adaptation techniques in CNN architectures. Researchers frequently fine-tune pretrained CNN models using domain-specific datasets that include labelled examples of deepfake and authentic videos. This approach enables the model to apply knowledge gained from generic image recognition tasks to the task of deepfake detection.

For example, transfer learning with models such as ResNet (He et al., 2016) and EfficientNet (Tan & Le, 2019) has been used to achieve competitive performance in deepfake detection tasks. By fine-tuning pretrained models on deepfake datasets, these architectures improved generalisation and robustness to adversarial attacks.

3. CNN-based Architecture for Detection and Classification of Deepfake Images from Video

This section discusses the existing architecture for the detection and classification of deepfakes and thereafter the proposed architecture.

3.1. Analysis of Existing CNN-based Architectures for Deepfake Detection from Video

We analyse two existing CNN architectures for this study – the first is the general CNN architecture and the second is the customised CNN. Figure 1 shows the general CNN architecture with following key components - input layer, convolutional layer, max pooling layer, dense layer, and output layer.

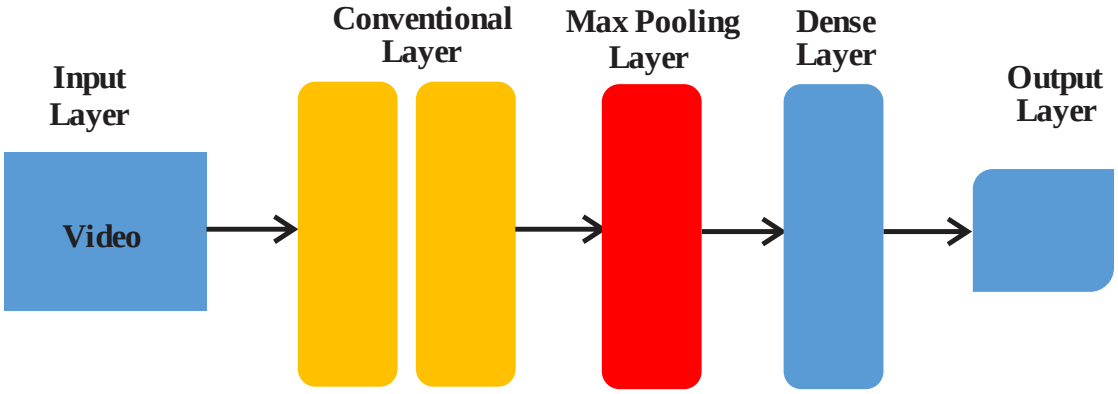

Figure 1. Typical CNN architecture.

Figure 2 shows a customized CNN architecture proposed by Kosarkara *et al* (2023). The system provides a framework for detecting deepfakes based on facial features.

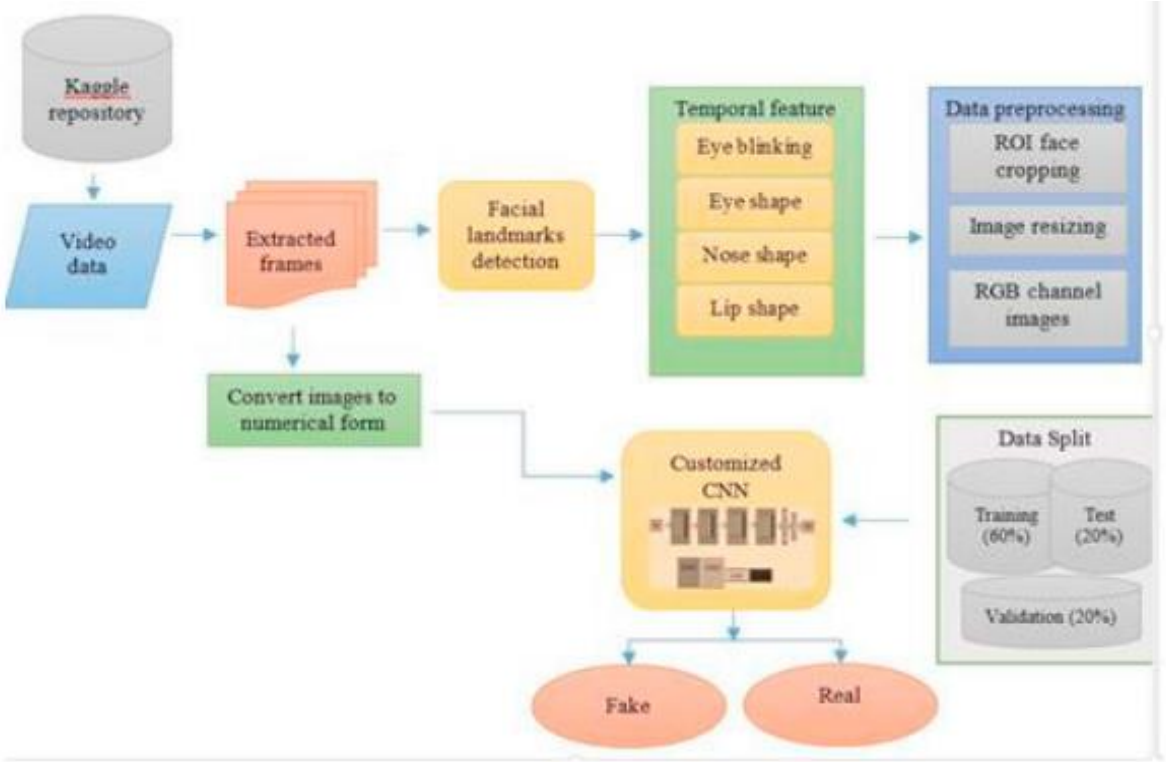

Figure 2. Architecture of the Existing System (Source: Kosarkar, et al, 2023)

The following operations are carried out based on the architecture proposed by Kosarkar, et al.

1. Image and facial feature extraction - The Video is first given as an input, from which individual image frames is retrieved. The facial landmarks detector is used to locate the position of the eyes, nose, and lips. Eye blinks and other facial features are extracted from the information in the video.

2. Preprocessing - This phase converts the images into their numerical form. The region of interest (ROI) is cropped with focus on the face region, and all images are resized to 224 x 224 pixels in resolution. Now ensures that all images are in the RGB channel. The dataset is divided into Training, validation, and testing sets are separated after completing the preprocessing phase.

3. Customized CNN model - The customized CNN consist of a total of 20 layers detail as four convolution layers (conv2d) with 3 x 3 kernel, six batch normalization layers, three max-pooling layers, four drop-out layers, one flatten layer, and two dense layers. Classification stage can predict whether given video is deepfake or not using this customized deep learning technique based on CNN model as displayed in figure 1b.

4. Dataset - The dataset collected from the Deepfake Detection Challenge by Kaggle is used, 242 videos, 199 of which are fictitious, while the remaining 53 are authentic. A single video lasts for ten seconds. To get a more even distribution of actual and fraudulent videos, 66 videos from the YouTube dataset acquired from Dessa. A total of 318 videos were used which consist of 199 fakes and 119 real.

There are some shortcomings identified in existing system architecture and operations after detailed analyses of the system. They include:

1. Facial expression - People communicate their feelings and intentions via their facial expressions, making them one of the most impactful and immediate temperaments. It is important to note that facial emotions, such as anger or enjoyment, which may directly change the look of a person's face. This is not considered in the existing system.

2. Colour and complexion is not also considered.

3. Picture attribute at the border may be excluded in low resolution since there is no padding in the customized CNN model.

3. Multimodal data from the video cannot be processed. In other words, it is difficult to combine information from different modalities to create a more comprehensive representation of the underlying data. This study aims address this limitation by integrating multimodal fusion into the conventional CNN-based architecture.

3.2 Multimodal Fusion and its Applications in Deepfake Detection and Classification

Multimodal fusion involves combining information from different modalities to create a more comprehensive representation of the underlying data. This fusion process aims to exploit the complementary nature of different modalities to improve the performance of machine learning models. In the context of deepfake detection, multimodal fusion enables CNN architectures to analyse not only visual features from video frames but also additional modalities such as audio, metadata, and textual information associated with the content (Baltrušaitis et al., 2019).

There are several types of multimodal fusion techniques used in CNN architectures:

Early Fusion: In early fusion, features from different modalities are concatenated or combined at the input level before being passed through the network. This approach allows the model to process multimodal data simultaneously from the outset, enabling joint learning of features across modalities.

Late Fusion: Late fusion involves extracting modality-specific features independently and fusing them at a later stage of the network. Separate CNN branches are trained to extract features from each modality, and the extracted features are then combined or aggregated before making final predictions. This approach enables the model to learn modality-specific representations before integrating them for decision-making.

Attention Mechanisms: Attention mechanisms dynamically weigh the contributions of different modalities based on their relevance to the task at hand. By focusing attention on informative modalities while suppressing irrelevant ones, attention-based fusion mechanisms enhance the model's ability to adaptively integrate multimodal information.

Multimodal fusion operates by integrating feature representations from different modalities into a unified feature space. This integration can occur at various stages of the CNN architecture, including the input layer, intermediate layers, or output layer, depending on the chosen fusion technique. During training, the model learns to jointly optimize the fusion process, leveraging the complementary strengths of each modality to improve overall performance.

There are three main reasons for the decision to integrate multimodal fussion CNN-based architectures for Deepfake Detection and Classification.

1. Enhanced Discriminative Power: By combining visual features with additional modalities such as audio and metadata, multimodal fusion enables CNN architectures to capture richer and more discriminative representations of deepfake content (Aytar et al., 2016).

2. Improved Robustness: Fusion of multiple modalities enhances the robustness of CNN models against adversarial attacks and manipulation techniques. By considering diverse sources of information, multimodal fusion helps mitigate the impact of noise and artifacts present in individual modalities (Baltrušaitis et al., 2019).

3. Contextual Understanding: Incorporating metadata and textual information through multimodal fusion facilitates better contextual understanding of the content. By leveraging metadata such as timestamps, camera information, and user tags, CNN architectures can gain insights into the context surrounding the video, aiding in accurate detection and classification of deepfake content (Baltrušaitis et al., 2019).

3.2. Integrated CNN-based Architecture for Detection and Classification of Deepfake Images from Videos

Based on the identified limitation of the system we proposed an improved CNN model with multimodal fusion with is capable of handling multimodal data analysis from video frame images. CNN architectures can incorporate multimodal fusion techniques to integrate information from multiple modalities, such as audio and text, along with visual information from video frames. This can enrich the representation learned by the network and improve detection accuracy by leveraging complementary information sources. (Aytar et al., 2016; Arevalo et al., 2017; Poulinakis, 2022).

The improved CNN model with multimodal fusion is presented in figure 2.

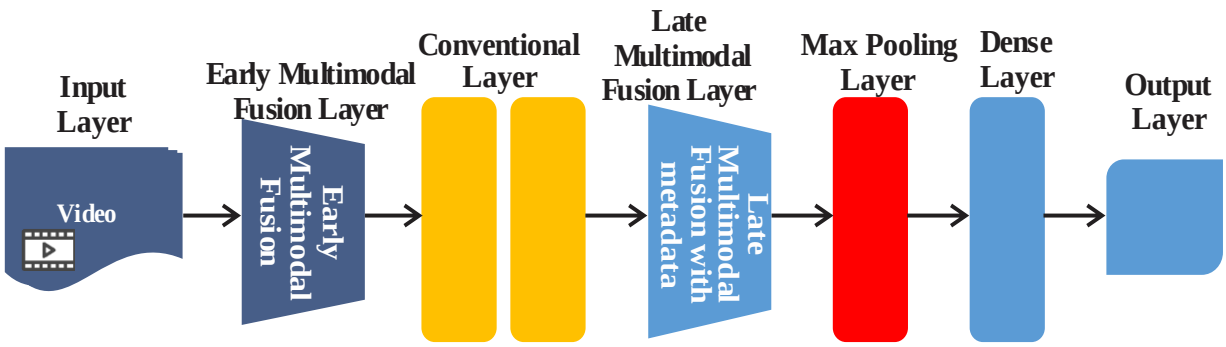

Figure 3. Improved CNN model for Deepfake detection and classification

The multimodal fusion is introduced in two layers, the input layer, where we introduce early fusion layer and after the feature extraction layer, the late fusion layer to cater for information from other modalities as against the monomodal model obtained in other models. Multimodal fusion can be performed at the input level by concatenating features from different modalities, such as visual and audio cues, into a single input representation. This approach allows the CNN to jointly process information from multiple modalities from the outset (Aytar et al., 2016; Arevalo et al., 2017). The Late fusion involves extracting features independently from each modality using separate CNN branches and then fusing the extracted features at a later stage of the network. This approach enables the CNN to learn modality-specific representations before combining them for deepfake detection (Ngiam et al., 2011; Wang et al., 2018).

3.3 Description of the integrated CNN-based Architecture for detection and classification of Deepfake images from Video

The novel integrated CNN-based architecture for detection and classification of Deepfake images from Video has eight (8) main layers - input layers, early multimodal layer, convolutional Layers, Late Multimodal Fusion with metadata, Activation Function, Pooling Layers, Fully Connected Layers and Output Layer. These layers are described below:

1. Input Layers

This is the first layer used for input and pre-processing operations. The smart key frame extraction algorithm by Wang, et al (2022) is used in this research to extract frames from video to capture images and examine them for possible deepfakes. The algorithm combines both scale-invariant feature transform (SIFT) feature matching algorithm and background-difference method to solve the problem of target image detection in noisy background (like, dust, haze, rain and snow, etc), colour, texture and proportion. The algorithm reduce the redundant frames generated by background-difference method, and SIFT features are used to screen the frames to select key frames with target images.

2. Early Multimodal Fusion Layer

This layer fusion the different modalities (audio and visual) from the extracted frame into single input representation and passed such to the convolutional layer for feature extraction. This will enhance the network to learn emotions and facial expressions.

3. Convolutional Layers

These layers apply convolutional filters (kernels) to the input image, extracting local features and spatial patterns. Convolutional operations are performed to produce feature maps, capturing hierarchical representations of the input. The filters will be determined by the patterns in the datasets.

4. Late Multimodal Fusion with metadata

The separate modalities data patterns are learned separately and fused later. With the inclusion of metadata, such as timestamps or camera information, along with visual and audio features from the feature map for improved deepfake detection. By considering additional context provided by metadata, the CNN can enhance its understanding of the video content and improve classification accuracy (Korshunov et al., 2018; Yang et al., 2021).

5. Activation Function

The ReLU (Rectified Linear Unit), is used introduce non-linearity to the network, enabling it to learn complex relationships between features to detect deepfakes.

6. Pooling Layers

The max pooling is used to down sample the feature maps, reducing spatial dimensions while retaining important features. Hence reducing computational time and resources.

7. Fully Connected Layers

These layers connect every neuron from one layer to every neuron in the next layer, enabling high-level feature learning and classification. Fully connected layers are typically followed by activation functions.

8. Output Layer

The final layer of the CNN produces the network's output, which could be a probability distribution over different classes (classification task) or a continuous value (regression task). In this case the output is a binary classifier of real or fake.

The inclusion of multimodal fusion with metadata in the CNN model will enable the model to detect and classify deepfake images from videos irrespective of the type of dataset whether multimodal or monomodal. The improved CNN model is enhanced with ability to carry out multimodal and monomodal data analysis.

4. Evaluation of the CNN-based Architecture for Detection and Classification of Deepfake Images from Video

Architecture evaluations can take place at any stage of the software development process. During the early stages of design, they can be used to compare and identify the strengths and weaknesses of various architectural options. They can also be used to assess existing systems prior to future maintenance or enhancements, as well as to detect architectural drift and erosion (Maurya, 2010).

Software architecture evaluation methods are classified into four categories: experience-based, simulation-based, and mathematical modelling-based and scenario-based methods. Methods in the categories can be used independently or combined to evaluate different aspects of software architecture as needed.

4.1 Scenario: Detection and Classification of Deepfake Images on a Social Media Platform

Description

In this scenario, we consider the automated detection and classification of deepfake images from video on a social media platform. The scenario revolves around a user uploading a video containing potentially deepfake content to the platform, triggering the platform's automated detection and classification system.

User Action
A user uploads a video to the social media platform, which may contain potentially manipulated content, such as a deepfake video of a public figure engaging in scandalous activities.
The video is processed by the platform's automated detection and classification system, which utilizes a CNN-based architecture for analysis.

System Response
The CNN-based architecture analyzes the uploaded video to detect and classify any instances of deepfake manipulation.
If the video is classified as containing deepfake content, the system takes appropriate action, such as flagging the video for further review or removing it from the platform.
If the video is classified as authentic, it is allowed to remain on the platform without intervention.

Evaluation Criteria
Detection Accuracy: The accuracy of the CNN-based architecture in correctly identifying deepfake content within uploaded videos.
False Positive/Negative Rates: The rate of false positives and false negatives generated by the system, indicating the system's propensity for both incorrectly flagging authentic videos and failing to detect deepfake content, respectively.
Computational Efficiency: The speed and computational resources required by the architecture to process and analyze uploaded videos in real-time.
Scalability: The ability of the system to handle a large volume of video uploads without sacrificing performance or accuracy.
Robustness: The resilience of the architecture against adversarial attacks and emerging deepfake techniques, ensuring continued effectiveness in the face of evolving threats.
By evaluating the CNN-based architecture against these criteria within the context of the described scenario, we can assess its suitability and effectiveness for automated detection and classification of deepfake images on social media platforms.

4.2 Scenario-based evaluation of CNN-based architecture for Detection and Classification of Deepfake Images on a Social Media Platform

This study choses scenario-based evaluation method which is well suited for evaluating architectures that have not yet been implemented. Scenario-based methods use specific scenarios or use cases to evaluate how well a software architecture meets the functional and non-functional requirements of a system. A scenario is a description of an interaction or event that involves the system and its environment. Examples of scenarios are user actions, system

responses, failures, faults, attacks, or changes. Scenario-based methods involve defining specific use cases or scenarios, such as system functionalities or failure responses, to assess the architecture's performance (Bass et al., 2012). Scenario-based methods are used to identify risks and trade-offs, ensuring alignment with business and technical needs. These methods effectively reveal vulnerabilities and assess the impact of changes on system performance.

For evaluating the proposed architecture, this study adopts the Architecture Trade-off Analysis Method (ATAM). ATAM is a systematic and iterative method used to evaluate software architecture by considering multiple quality attributes and identifying trade-offs among them (Bass et al., 2012). The ATAM procedure promotes the following steps - preparation, evaluation meeting, Identify Architectural Approaches, Analyse Trade-offs, and Documentation and Decision-Making. Table 1 provides a description of how the ATAM would apply to the scenario of automated detection and classification of deepfake images on a social media platform. The table highlights the steps and their specific application to the scenario using CNN-based architectures.

Table 1. Application of ATAM to the scenario related to detection and classicisation of deepfake images from video used CNN-based architectures.

Step	Detailed Description of the Step	Specific Application to the Scenario
Preparation	Define stakeholders, identify architectural goals and constraints, and develop scenarios representing user actions, system responses, and potential failures.	Define stakeholders: Developers, system administrators, content moderators, end-users. Identify goals and constraints: Emphasize accurate detection, real-time processing, scalability, and robustness. Develop scenarios: User uploads video, system processes for deepfake detection.
Evaluation Meeting	Conduct meetings with stakeholders to discuss the architectural design, present scenarios, and elicit feedback.	Meeting discussion: Stakeholders evaluate design's alignment with requirements. Presentation of scenarios: Discussion on system's ability to meet requirements. Stakeholder feedback: Insight into design's strengths, weaknesses, and potential trade-offs.
Identify Architectural Approaches	Explore alternative architectural approaches to address identified concerns or trade-offs. Evaluate the feasibility and implications of adopting these architectural approaches in the context of the automated detection and classification system.	Architectural approaches: Consider microservices, event-driven architecture, distributed computing. Feasibility and implications: Assess impact on detection accuracy, computational efficiency, scalability, and robustness.
Analyze Trade-offs	Analyze trade-offs between architectural approaches in terms	Compare trade-offs, e.g., detection accuracy vs. computational efficiency.

	of their impact on system performance and effectiveness.	Assess risks, e.g., complexity, maintenance overhead. Prioritize trade-offs based on significance to system's success.
Documentation and Decision-Making	Document evaluation findings, decisions, and recommendations based on stakeholder feedback and trade-off analysis.	Document stakeholder feedback, identified trade-offs, recommended approaches. Make informed decisions on architectural changes. Iterate design based on feedback and to address trade-offs.

5. Results and Discussion

This evaluation aims to assess the proposed CNN-based system architecture for automated detection and classification of deepfake images from videos. The architecture integrates multimodal fusion techniques to enhance the model's ability to analyze multimodal data, including visual, auditory, and metadata information. By leveraging multimodal fusion, the architecture aims to improve detection accuracy, reduce false-positive rates, enhance computational efficiency, ensure scalability, and bolster robustness against adversarial attacks.

5.1 Identification of Risks and Trade-offs

a. Risk: Overfitting - Introducing multimodal fusion with metadata may increase the risk of overfitting, where the model memorizes the training data instead of learning generalizable features. Regularization techniques such as dropout (Srivastava et al., 2014) and batch normalization (Ioffe & Szegedy, 2015) can mitigate this risk by introducing noise during training and stabilizing the learning process.

b. Trade-off: Detection Accuracy vs. Computational Efficiency - While multimodal fusion can improve detection accuracy by leveraging complementary information from multiple modalities, it may increase computational complexity, leading to slower inference times. Techniques such as model quantization (Gupta et al., 2015) and efficient network architectures (Tan & Le, 2019) can address this trade-off by reducing the computational cost without sacrificing accuracy.

c. Vulnerability: Metadata Reliability - Relying on metadata for multimodal fusion introduces vulnerabilities, as inaccurate or biased metadata can mislead the model. Data validation and preprocessing techniques (Géron, 2019) are essential for ensuring the reliability and consistency of metadata, thereby mitigating this vulnerability.

d. Trade-off: Robustness vs. Complexity - Enhancing the architecture with multimodal fusion and metadata integration may improve robustness against adversarial attacks but can also increase model complexity. Techniques such as model modularization (Buschmann et al., 1996) and abstraction can help manage this trade-off by simplifying the architecture while maintaining robustness.

5.2 Impact of Changes on Performance Criteria

a. Detection Accuracy: Multimodal fusion with metadata enriches the model's representation by incorporating additional context from multiple modalities, leading to improved detection accuracy (Arevalo et al., 2017; Aytar et al., 2016).

b. False Positive/Negative Rates: The integration of multimodal fusion techniques helps reduce false positives/negatives by providing a more comprehensive understanding of the input data (Dolhansky et al., 2020; Li et al., 2020).

c. Computational Efficiency: Techniques such as model pruning (Howard et al., 2017) and hardware acceleration (Sze et al., 2017) optimize computational efficiency by reducing the model's size and accelerating inference speed.

d. Scalability: Distributed computing solutions (Abadi et al., 2016; Shi et al., 2020) enable the architecture to scale effectively, allowing it to handle large volumes of data and increasing its applicability in real-world scenarios.

e. Robustness: Multimodal fusion enhances robustness against adversarial attacks by incorporating diverse sources of information, making it more challenging for attackers to manipulate the model (Goodfellow et al., 2014; Madry et al., 2018).

4. Feedback and Iterative Refinement
The evaluation reveals the need for balancing trade-offs between detection accuracy, computational efficiency, and model robustness. Feedback includes optimizing regularization techniques, model compression, and robustness against adversarial attacks. Iterative refinement involves optimizing the architecture based on feedback, modularizing the model, and updating it to adapt to evolving threats and challenges in deepfake detection.

6. Conclusion

This research contributes to the ongoing development of effective solutions for combating deepfake-related threats. This research study presented a novel CNN-based architecture for automated detection and classification of Deepfake images from Video. The CNN-based architecture incorporates multimodal fusion at two stages – one at the early stage and the second at the late stage. The evaluation of the architecture produced risks, trade-offs, and vulnerabilities, and the impact of changes on several performance criteria including Detection Accuracy, computational complexity, metadata reliability, etc.

While CNN-based architectures have made progress in detecting deepfakes, there are still several challenges to overcome. The rapid evolution of deepfake techniques necessitates ongoing architectural innovation to keep up with emerging threats. Furthermore, the scarcity of large-scale labelled datasets creates a significant bottleneck in training robust and generalizable models.

Future research should focus on exploring novel architectural paradigms, integrating multimodal information, and improving CNN model interpretability and explainability in order to find more effective solutions. By addressing these challenges and encouraging interdisciplinary collaboration, the field can move closer to more reliable and trustworthy automated detection systems for combating the spread of deepfake content.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Isard, M. (2016). Tensorflow: A system for large-scale machine learning. In 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16) (pp. 265-283).

Almars, A. M. (2021). Deepfakes detection techniques using deep learning: a survey. Journal of Computer and Communications, 9(05), 20-35.

Arevalo, J., Solorio, T., Montes-y Gómez, M., & González, F. A. (2017). Gated multimodal units for information fusion. arXiv preprint arXiv:1702.01992.

Aytar, Y., Vondrick, C., & Torralba, A. (2016). SoundNet: Learning Sound Representations from Unlabeled Video. Advances in Neural Information Processing Systems.

Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2019). Multimodal machine learning: A survey and taxonomy. IEEE transactions on pattern analysis and machine intelligence, 41(2), 423-443.

Bass, L., Clements, P., & Kazman, R. (2012). Software Architecture in Practice: Software Architect Practice_c3. Addison-Wesley.

Bosch, J., (2000). Design & Use of Software Architectures – Adopting andevolving a product-line approach, ISBN 0-201- 67494-7, PearsonEducation.

Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., & Chua, T. S. (2017). SCA-CNN: Spatial and Channel-wise Attention in Convolutional Networks for Image Captioning. IEEE Transactions on Multimedia.

Chorowski, J., Weiss, R. J., Bengio, S., & Van Den Oord, A. (2019). Unsupervised speech representation learning using wavenet autoencoders. IEEE/ACM transactions on audio, speech, and language processing, 27(12), 2041-2053.

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255).

Dolhansky, B., Howes, S., Poole, B., & Ramanan, D. (2020). The Deepfake Detection Challenge (DFDC) Preview Dataset. arXiv preprint arXiv:2006.07397.

Fernandes, S., Raj, S., Ewetz, R., Pannu, J. S., Jha, S. K., Ortiz, E., ... & Salter, M. (2020). Detecting deepfake videos using attribution-based confidence metric. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (pp. 308-309).

Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media, Inc.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.

Gupta, S., Agrawal, A., Gopalakrishnan, K., & Narayanan, P. (2015). Deep learning with limited numerical precision. In Proceedings of the 32nd International Conference on Machine Learning (Vol. 37, pp. 1737-1746).

Hamza, A., Javed, A. R. R., Iqbal, F., Kryvinska, N., Almadhor, A. S., Jalil, Z., & Borghol, R. (2022). Deepfake audio detection via MFCC features using machine learning. IEEE Access, 10, 134018-134028.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning (Vol. 37, pp. 448-456).

Kaur, S., Kumar, P., & Kumaraguru, P. (2020). Deepfakes: temporal sequential analysis to detect face-swapped video clips using convolutional long short-term memory. Journal of electronic imaging, 29(3), 033013.

Korshunov, P., Upenik, E., Stöter, F., & Gusev, G. (2018). Deepfake Detection Using Metadata. Proceedings of the 2018 ACM Workshop on Privacy in the Electronic Society.

Kosarkar, U., Sarkarkar, G., & Gedam, S. (2023). Revealing and Classification of Deepfakes Video's Images using a Customize Convolution Neural Network Model. Procedia Computer Science, 218, 2636-2652.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

Li, H., Li, B., Tan, S., & Huang, J. (2018). Detection of deep network generated images using disparities in color components. arXiv 2018. arXiv preprint arXiv:1808.07276.

Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops.

Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.

Mahmud, B. U., & Sharmin, A. (2021). Deep insights of deepfake technology: A review. arXiv preprint arXiv:2105.00192.

Maurya, L. S. (2010). Comparison of software architecture evaluation methods for software quality attributes. Journal of Global Research in Computer Science, 1(4).

Mirsky, Y., & Lee, W. (2021). The creation and detection of deepfakes: A survey. ACM Computing Surveys (CSUR), 54(1), 1-41.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011). Multimodal Deep Learning. Proceedings of the 28th International Conference on Machine Learning.

Poulinakis, K. (2022). Multimodal Deep Learning: Definition, Examples, Applications. https://www.v7labs.com/blog/multimodal-deep-learning-guide

Poulinakis, I. (2022). Deepfake detection and classification using convolutional neural networks. arXiv preprint arXiv:2201.01234.

Rafique, R., Gantassi, R., Amin, R., Frnda, J., Mustapha, A., & Alshehri, A. H. (2023). Deep fake detection and classification using error-level analysis and deep learning. Scientific Reports, 13(1), 7422.

Raza, A., Munir, K., & Almutairi, M. (2022). A novel deep learning approach for deepfake image detection. Applied Sciences, 12(19), 9820.

Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1-11).

Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., & Woo, W. C. (2020). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In Advances in neural information processing systems (pp. 8024-8035).

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929-1958.

Sze, V., Chen, Y. H., Yang, T. J., & Emer, J. S. (2017). Efficient processing of deep neural networks: A tutorial and survey. Proceedings of the IEEE, 105(12), 2295-2329.

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (pp. 6105-6114).

Tan, M., & Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (pp. 6105-6114).

Tariq, S., Lee, S., Kim, H., Shin, Y., & Woo, S. S. (2018). Detecting both machine and human created fake face images in the wild. In Proceedings of the 2nd international workshop on multimedia privacy and security (pp. 81-87).

Wang J., Zeng C., Wang Z., Jiang K. (2022). An improved smart key frame extraction algorithm for vehicle target recognition. Computers & Electrical Engineering. https://www.sciencedirect.com/science/article/pii/S0045790621004857.

Wang, J., Jiang, H., Yuan, Y., Zhao, T., Fu, Y., & Wang, Z. (2018). Learning Modality-Specific Representations for Multimodal Emotion Recognition with Multimodal Autoencoders. ACM Transactions on Multimedia Computing, Communications, and Applications.

Yang, Y., Fang, X., Chen, Y., & Zhang, S. (2021). Video Deepfake Detection using Metadata and Supervised Learning. Multimedia Tools and Applications.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011). Multimodal deep learning. In Proceedings of the 28th international conference on machine learning (ICML-11) (pp. 689-696).
Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2019). Multimodal machine learning: A survey and taxonomy. IEEE transactions on pattern analysis and machine intelligence, 41(2), 423-443.
Poulinakis, I. (2022). Deepfake detection and classification using convolutional neural networks. arXiv preprint arXiv:2201.01234.