IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 January 2023, accepted 21 February 2023, date of publication 2 March 2023, date of current version 8 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3251417

RESEARCH ARTICLE

An Improved Dense CNN Architecture for Deepfake Image Detection

YOGESH PATEL[1], SUDEEP TANWAR[1], (Senior Member, IEEE), PRONAYA BHATTACHARYA[2], (Member, IEEE), RAJESH GUPTA[1], TURKI ALSUWIAN[3], INNOCENT EWEAN DAVIDSON[4], (Senior Member, IEEE), AND THOKOZILE F. MAZIBUKO[5]

[1]Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, Gujarat 382481, India
[2]Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University, Kolkata 700135, India
[3]Electrical Engineering Department, College of Engineering, Najran University, Najran 11001, Saudi Arabia
[4]Department of Electrical, Electronic and Computer Engineering, Cape Peninsula University of Technology, Bellville 7535, South Africa
[5]Department of Electrical Power Engineering, Durban University of Technology, Durban 4001, South Africa

Corresponding authors: Thokozile F. Mazibuko (ThokozileM1@dut.ac.za), Sudeep Tanwar (sudeep.tanwar@nirmauni.ac.in), Pronaya Bhattacharya (pbhattacharya@kol.amity.edu), and Rajesh Gupta (rajesh.gupta@nirmauni.ac.in)

ABSTRACT Recent advancements in computer vision processing need potent tools to create realistic deepfakes. A generative adversarial network (GAN) can fake the captured media streams, such as images, audio, and video, and make them visually fit other environments. So, the dissemination of fake media streams creates havoc in social communities and can destroy the reputation of a person or a community. Moreover, it manipulates public sentiments and opinions toward the person or community. Recent studies have suggested using the convolutional neural network (CNN) as an effective tool to detect deepfakes in the network. But, most techniques cannot capture the inter-frame dissimilarities of the collected media streams. Motivated by this, this paper presents a novel and improved deep-CNN (D-CNN) architecture for deepfake detection with reasonable accuracy and high generalizability. Images from multiple sources are captured to train the model, improving overall generalizability capabilities. The images are re-scaled and fed to the D-CNN model. A binary-cross entropy and Adam optimizer are utilized to improve the learning rate of the D-CNN model. We have considered seven different datasets from the reconstruction challenge with 5000 deepfake images and 10000 real images. The proposed model yields an accuracy of 98.33% in AttGAN, [Facial Attribute Editing by Only Changing What You Want (AttGAN)] 99.33% in GDWCT,[Group-wise deep whitening-and-coloring transformation (GDWCT)] 95.33% in StyleGAN, 94.67% in StyleGAN2, and 99.17% in StarGAN [A GAN capable of learning mappings among multiple domains (StarGAN)] real and deepfake images, that indicates its viability in experimental setups.

INDEX TERMS Deepfake detection, CNN, convolutional neural network, GAN.

I. INTRODUCTION

Artificial intelligence (AI) has progressed in diverse domains, including computer vision, speech generation and analysis, and the design of multi-agent systems in the industry. In a similar direction, generative deep learning (DL) techniques have made a transformative shift in multimedia processing, where recently, deepfakes (DF) have emerged, which allows the creation of synthetic content based on captured images and videos of persons. In DF, a person's eyes, lips, and face movements are captured and superimposed on another external environment that forms a realistic vision of that person in a simulated fake environment. With the world becoming more connected and networked through social media circles, DFs are increasingly used to create synthetic data of politicians, communities, actors, and media that give rise to fake news generation and dissemination. To generate DFs, one effective algorithm is generative adversarial networks

The associate editor coordinating the review of this manuscript and approving it for publication was Jeon Gwanggil.

(GANs), initially proposed by Goodfellow et al. in 2014 [1], [2]. GANs [3] make it easy to create a fake synthetic image, audio, and video content presented as real.

Technically, the GAN model comprises two networks: a generator network (which aims to generate synthetic content out of the noise vector) and a discriminator network (which aims to classify these generated synthetic images). An iterative process is followed in the generator-discriminator network, where the discriminator feedback is supplied to the generator network. Over time, the generator learns to create synthetic content, which looks extremely real and spoofs the discriminator [4], [5]. Thus, the generator-discriminator network in DF raises concerns about the authenticity of the published content on social platforms, as it is tough to differentiate between real and fake content. Some notable examples of DF include tools like DFaker, DeepFaceLab, Faceswap, Faceswap-GAN, STGAN, StarGAN, and Face Swapping GAN (FSGAN), and many others [6]. In DeepFaceLab, it allows a user to swap a person's face with another person's face, change the age of a person, and synchronize the lip and eye movements in the video [7]. Face2Face [8] allows a real-time face enactment based on the RGB video output, and the emulation of input expressions is carried out. DF tools are also used in generating pornographic content that hurts the sentiments of the public [9], [10]. However, hate speeches are other widely used propaganda in social circles. For example, a video of former United States 44^{th} President Barack Hussein Obama II published by *BuzzFeed* shows the former president cursing another former president Donald Trump, which is done through the GAN technology. It is massively distributed in social media circles as official news, but the content is synthetic [11].

Thus, it raises a prime concern about the authenticity of news content. To overcome the aforementioned issues of DF GANs, a robust and highly generalizable DF detection system is required. A good DF detection system can detect highly accurate manipulated and synthetic content from authentic content. Recent approaches published in the literature point to the design of a robust DF detection scheme. Most of the approaches in the literature lack robustness, effectiveness in training the DF detection model, and integration of generalizability and interpretability in the model [12], [13], [14]. As indicated by Yu et al. in [12], the robustness in the DF detection means that the system should be able to detect manipulation of high-quality and low-quality image/video contents. The system's effectiveness should not be dropped based on the resolution of contents. Generally, the performance of DF detection systems drops over low-quality content. Generalizability refers to the condition where each DF generation tool utilizes different approaches to generate the DF contents. Thus, the DF system should be able to detect manipulations from these different tools in a single-shot [12]. Interpretability refers to the condition in the DT detection ecosystem, where a model should be able to predict which parts of the image (person's face, for example) are real or fake and label the bounding boxes with fake probabilities. Thus, it is crucial as it enables a system to understand the dynamics of generated synthetic content and presents a visual explanation to understand the abnormalities in the images [15]. Current systems analyze DF detection on a sequential frame-by-frame basis, which results in higher temporal inconsistencies in the model. Thus, there is a stringent requirement for effective DF detection models that can form an optimal mix of the aforementioned conditions [16].

Recent approaches have suggested convolutional neural networks (CNN) as an effective fit for DF detection models [17]. Usually, pre-trained CNN models are applied on single frames, while other approaches have considered recurrent convolutional networks where frames can be grouped to form the decision. In addition, some approaches consider facial expression patterns to capture fake content. Most CNN-based approaches are black boxes, where the models are overfitting. In other cases, the validation, testing, and training split are not uniformly distributed, which leads to different interpretations of the same datasets under different operating conditions. For example, a DF detection model on the Facebook DF detection challenge dataset is proposed [18]. The model scored an average precision of 82.56% on these datasets, but the performance drastically drops to 65.18% on the validation dataset, as it is collected from various sources. Thus, a generalization through CNN on one dataset does not hold a cross-performance on another dataset [19]. The inconsistencies can be mitigated through an effective deep CNN (D-CNN) model that can address the cross-domain interpretability while maintaining the robustness and generalizability of the DF detection scheme, which would yield a high accuracy through an effective ensemble to the proposed CNN approaches.

A. NOVELTY

Existing CNN-based DF detection models should conform to the abilities of high generalizability, robustness, and interpretability [12], [20]. The lack of the above-mentioned abilities can be seen in the existing systems such as MesoNet, MesoInceptionNet and many others. These are some well-known CNN-based compact DF detection models focused on detecting deepfake images for low-quality images. Even though yielding promising results on the test set, these models lack generalizability capabilities which is a well-discussed challenge in the domain of DF detection. Accuracy drops by a huge margin whenever these DF detection methods are tested against DF images generated using different methods. DF detection systems learn certain features particular to the generation methods whose images were used to train these models. For Example, if any DF detection model was developed and trained over images from StarGAN and then tested over reserved unseen test images will definitely yield good results, but when tested against images from some other DF generation method, say STYLEGAN, then accuracy will drop by a huge margin. Sometimes accuracy drops

to the point that it becomes just a random guess from the model. Hence indicating a lack of generalizability being the challenge at large. Regardless, CNN approaches have mostly treated DF detection as a binary classification problem, where cross-domain interoperability is required [18]. The proposed work presents an improvement over MesoNet and MesoInceptionNet, a D-CNN model that extracts deep features from input images through the convolution layer to address the aforementioned challenges. It captures the manipulation traces left behind as features and forms a classification model based on the similarities between real and fake images. The similarities are projected to the closest match that improves the model predictability, as it captures the complex inconsistencies through the deep network. Furthermore, the model is trained over synthetic and real images from different sources, improving the generalizability and cross-learning accuracy.

B. RESEARCH CONTRIBUTION

Following are the major contributions of the paper.

- We analyze various existing approaches to DF detection using the CNN model and highlight their advantages and potential pitfalls.
- We propose a novel D-CNN-based architecture to classify DF image and video contents. The proposed model is trained over images from seven data sources to increase its generalizability.
- We then evaluate the performance of the proposed architecture using accuracy, precision, recall and F1 score metrics over the reserved test set.

C. ARTICLE LAYOUT

The layout of the article is as follows. Section II presents the existing approaches of DF detection models. Section III presents the problem formulation of the proposed DF classification scheme. Section IV details the proposed model approach and the systematic explanation of the model processing. Section V discusses the performance evaluation of the model based on various metrics. Section VI presents the discussion and future challenges in the proposed scheme, and finally, section IX presents the work's conclusion and future scope.

II. RELATED WORKS

From the literature, it can be seen that researchers have already adopted different types of approaches to create an efficient DF detection system. Even though the approaches are there, their underlying principles in most approaches remain consistent, focusing on the utilization of inconsistencies and manipulation traces left behind by GAN tools during the generation network [6]. Although nowadays, DF spans multiple modalities such as audio, video, image, or hybrid modality-based models. Among these, the image/video-based DF is the most prominent; thus, most research is directed toward identifying image and video DFs. Thus, the image/video DF detection models are generally classified into three domains: physical/physiological features, signal level features, and data-driven models [21]. DF detection approaches involve more than one modality, i.e., combined audio and video is termed multi-modal approach, where the classification rests on computing the disharmony (or entropy difference) between two different modalities in DF manipulations [22], [23]. Table 1 presents a comparative analysis of the proposed D-CNN model against existing approaches in terms of approach and the proposed method. The following subsection presents the existing approaches in the classified domains.

A. PHYSICAL/PHYSIOLOGICAL FEATURES

In physical and physiological feature-based approaches, visible discrepancies in the image or video content are exploited to classify whether the submitted content is synthetic or real. The visible discrepancies primarily include improper shadows, irregular geometry, missing details in facial features such as teeth or ears, inconsistent eye colors, head movements, and other features. For example, Li et al. [24] leveraged inconsistencies in the blinking eye patterns, which the DF tools cannot mimic in a video stream. Authors in [28] worked on the inconsistencies in the head pose movements compared to the rest body movements in the DF image and videos and identified the synthetic content in the data. The authors identified 68 different landmarks in the whole body, which includes 17 facial landmarks on the face. The direction movement is considered from the center of the face, and if the directions on two or more landmarks are the same, then they are classified as authentic content or synthetic. Matern et al. [29] tried to use inconsistencies in other visual artifacts such as inconsistent geometry of teeth, shadow, lighting, and eye colors. However, the considered approach is good, but the latest DF generation tools have exploited and learned about the geometry of faces, and thus the said models can easily spoof the model. Thus, to overcome the feature-based inconsistencies, the authors shifted to other representations, including signal-level feature extraction.

B. SIGNAL-LEVEL FEATURES

In signal-level features, deep features are extracted using either feature descriptors or feature extraction algorithms. Thus, low-level features are extracted using steganalysis, which the classification algorithm can use to classify whether the input content is DF. Kharbat et al. [30] presented a combination model of different signal-level feature descriptions based on HOG, ORB, SURF, and others. The extracted deep features are then fed as input to the SVM classifier to find whether the image is DF. Authors in [34] utilized a feature extraction approach known as scale-invariant feature transform (SIFT), which extracts key pixel features and analyzed them. Similar to the study of [30], Akhtar et al. [31] used local image descriptors such as LBP, LPQ,

TABLE 1. A comparative analysis of the proposed model with the existing approaches.

Author	Year	Approach	Algorithm	Method	Remarks
Li *et al.* [24]	2018	Physical attributes-based detection	Long-term recurrent CNNs	Used eye blinking pattern to detect DF videos	Advanced DF videos are hard to detect using visual feature sets
Marra *et al.* [25]	2018	Data-driven models	XceptionNet	Performed a comparative study of InceptionNet, DenseNet, and XceptionNet models. Among these, XceptionNet performed best	Lack of generalizability
Hsu *et al.* [26]	2018	Data driven models	CNN	Proposed a five layer CNN architecture called Deep Forgery Discriminator	Provides good results but lacked generalizability
Afchar *et al.* [20]	2018	Data-driven models	CNN	It is a CNN model which utilizes inception module as architecture backbone	Worked well with compressed videos, but XceptionNet outperformed on every dataset
Guera *et al.* [27]	2018	Data-driven models	RNN	RNN-based temporal feature model	accuracy is not effectively high and can be outperformed via other models
Yang *et al.* [28]	2019	Physical attributes-based detection	Support vector machine (SVM) classifier	Exploited inconsistencies between the head pose of the face and other parts of the body using various facial landmarks	Visual features are not reliable with advanced DF datasets
Matern *et al.* [29]	2019	Physical attribute-based detection	Ensemble model with multi-layer perceptron and logistic regression	Used visual artifacts, such as difference in eye colors, disproportionate shadow, details of invisible light reflections, and shape geometry	Visual features are not reliable with advanced DFs
Kharbat *et al.* [30]	2019	Signal level feature-based detection	SVM classifier	Combined multiple feature-point-descriptors, such as histogram of oriented gradients (HOG), features from accelerated segment test (FAST), binary robust independent elementary features (BRIEF), binary robust invariant scalable keypoints (BRISK), KAZE, speeded-up robust features (SURF), and oriented FAST and rotated BRIEF (ORB). HOG achieved an accuracy of 94.5% with the SVM classifier	With advanced DF coming up every year, extracting features is getting difficult.
Akhtar *et al.* [31]	2019	Signal level feature-based detection	SVM classifier	Used local image descriptors, such as local binary pattern (LBP), local phase quantization (LPQ), pyramid histogram of oriented gradients (PHOG), binary gabor pattern (BGP), and image quality metric (IQM).	IQM performed best among other models.
Nguyen *et al.* [32]	2019	Data-driven models	Capsule network	The capsule network consists of 3 primary capsules and 2 output capsules. Features extracted from VGG-19 are provided as input.	It worked as good as MesoNet, but XceptionNet outperforms all the networks
Amerini *et al.* [33]	2019	Data-driven models	CNN	Exploited discrepancies in motion across successive frames at $f(t)$ and $f(t+1)$. Used CNN as a classification algorithm	Other algorithms outperforms the proposed model
Proposed Model	2022	Data-driven model	CNN	Proposed D-CNN based architecture trained over images from seven different data sources	Data pipeline in the proposed architecture over DF videos

PHOG, SURF, BSIF, and IQM. The results suggested that IQM performed more accurately than other models. However, as DF tools became more sophisticated, the GAN model fooled signal-level feature descriptors. Thus, the research shifted towards the data-driven DF detection models.

C. DATA-DRIVEN MODELS

In data-driven DF detection, we use deep neural networks (DNN) instead of specific features to extract and learn about the feature. Based on the learning, the model classifies the submitted content as DF or real images/videos. However, to train the DNN model, a sufficient amount of data must be supplied to the model, and thus the approach is named data-driven. Marra et al. [25] used networks such as InceptionNet, DenseNet, and XceptionNet, with a large dataset of samples collected from different categories from image-to-image translation, which were created using CycleGAN. The results of their experiments suggested that XceptionNet outperforms all the other networks considered in the study. However, the issue of generalizability remains, which was addressed by the authors in [26], where they proposed a deep forgery discriminator network, which is essentially a five-layer CNN architecture based on embedding the contrastive loss. The results were promising, but lack of generalizability remains the praoblem. Another CNN-based approach is proposed by Afchar et al. [20], known as MesoNet, and it performed well as it focused on the mesoscopic features of the images. Nguyen et al. [32] proposed a capsule network with features extracted from VGG-19. The model performed as well as MesoNet, but XceptionNet still outperforms it. Similar approaches are present, where the authors used the temporal component of the video to identify DF videos.

Guera et al. [27] proposed a recurrent neural network (RNN) model, and Amerini et al. [33] used CNN with the concept of using discrepancies across frames to identify DF videos.

As outlined in the literature review section, the data-driven models normally outperformed the physiological and signal-based approaches. Thus, we consider a data-driven approach in the proposed scheme and propose a D-CNN model that captures the deep features with improved generalization and model predictability.

III. PROBLEM FORMULATION

This section presents the problem formulation of the proposed approach. The proposed model is a data-driven D-CNN model for DF detection that predicts the respective class of input images based on their features. To formulate the problem, we consider a certain amount of available images, represented as $I_{total} = \{I_1, I_2, \ldots, I_n\}$. I_{total} are sent for training and are classified into real images, represented as I_r or DF images, represented as I_{df}. I_r is constructed from p different data sources of real images, where any image $i \in I_r$ is represented as follows:

$$I_r = \left[\mathbf{N}_{i=1}^{k=1}, \mathbf{N}_{i=2}^{k=1}, \ldots, \mathbf{N}_{i=x}^{k=1}, \mathbf{N}_{i=1}^{k=2}, \ldots, \mathbf{N}_{i=x}^{k=2}, \ldots, \mathbf{N}_{i=x}^{k=p}\right] \tag{1}$$

Considering, each data source consists of x real images, it can be denoted as N_k. Thus, I_r is further denoted as follows:

$$I_r = \sum_{k=1}^{p} N_k \tag{2}$$

Similarly, for DF images, I_{df}, there are q data sources of deepfake images, and each source consists z images. The same is illustrated as follows:

$$I_{df} = \left[\mathbf{N}_{i=1}^{j=1}, \mathbf{N}_{i=2}^{j=1}, \ldots, \mathbf{N}_{i=z}^{j=1}, \mathbf{N}_{i=1}^{j=2}, \ldots, \mathbf{N}_{i=z}^{j=2}, \ldots, \mathbf{N}_{i=z}^{j=q}\right] \tag{3}$$

Similar to equation (2), we assume that z images is denoted as a set N_j. Thus, I_{df} is represented as:

$$I_{df} = \sum_{k=1}^{q} N_j \tag{4}$$

Based on equation (2), and equation (4), I_{total} is described as follows:

$$I_{total} = I_r + I_{df} = \sum_{k=1}^{p} N_k + \sum_{k=1}^{q} N_j \tag{5}$$

The labels for the corresponding classes can be defined as:

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m] \tag{6}$$

where m = total number of images in I_{total}. The proposed architecture comes under the binary classification problem, where there are only two classes, i.e., $y = 0$ indicating I_r, and $y = 1$ indicating I_{df} image.

Algorithm 1 Working of the Proposed Approach

Input: I - RGB Images of Face, D - Destination address of stored images, M - Destination address of pretrained model
Output: L - predicted likelihood, $\mathcal{P}$ - predicted label

```
procedure Deepfake_Detection( )
    Ht (Height) ← 160
    Wt (Width) ← 160
    DataGen ← ImageDataGenerator()
    Generator ← DataGen.flow_dir(D, Ht, Wt)
    model ← load_model(M)
    i ← 1
    while i ≤ len(Generator.labels) do
        I ← Generator.next()
        L ← model.predict(I)
        P ← round(L)
        Display likelihood L
        Display predicted label P
        Display Image I
        i ← i+1
    end while
end procedure
```

IV. PROPOSED APPROACH

As discussed in Section III, the proposed model is a binary classification model, where the input I_{total} is classified into I_r or I_{df} classes from multiple data sources for each class. For DF detection, CNN is a prominent choice. Thus, we augment the CNN model with a deep layer and present the D-CNN model to extract the deep features from input images using convolutional layers. Convolution operations performed over the images in the earlier stages allow us to extract much deeper features that can be used to classify the input images into DF. Figure 1 presents the details of our proposed model.

A. ALGORITHM

Algorithm 1 shows the proposed architecture's flow. It takes the facial image as input and its directory address (where the image is stored). It outputs the predicted likelihood and predicted label and prints the image that has been processed. The procedure algorithm follows that it sets the *Height* and *Width* to be 160. Then an object of *ImageDataGenerator* will be created with all the necessary arguments for the required data augmentation techniques. This created object is called *DataGen*. Using this *DataGen* object, we can flow the images one by one or in batches based upon the arguments given to *flow_from_directory()*. This *flow_from_directory()* accepts the destination address of the stored images, *Height* and *Width* as arguments. It will resize the input image to a given size and apply all the data augmentation techniques when required. It returns an object called *Generator*. Now the pretrained model must be loaded in an object called *model*. Users can one by one read an image from the input directory and give it as input to the model to predict the likelihood of

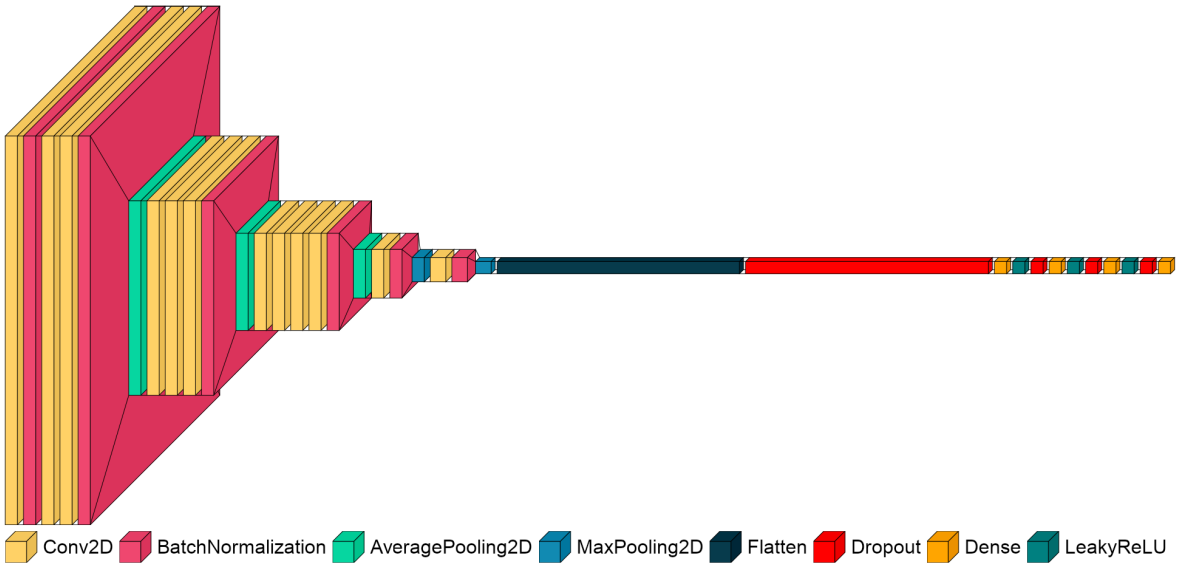

FIGURE 1. The proposed D-CNN model.

the prediction. Rounding up the predicted likelihood gives the predicted label of the class, and at the end, the image is also printed along with these two outputs.

The predicted likelihood ranges from 0 to 1. The closer it is to zero, the more confident mode is that the image is real. Vice versa, the closer likelihood is to 1, the confident model is about being the image deepfake. The closer it is to 0.5, it is much like a random guess. And thus, rounding of the predicted likelihood gives the predicted label. Real is indicated by '0', and Deepfake is by '1'. The image is also printed alongside these results for the user.

B. PROPOSED ARCHITECTURE

This section discusses the proposed CNN-based architecture (Figure 2). In General, CNN architecture consists of both convolutional and pooling layers. Convolutional layers extract deep features from input images, whereas pooling layers reduce the dimensionality of the input feature maps. After convolutional layers, all these feature maps are made into a one-dimensional array using a flattened layer and given as input to the fully connected layer. After the fully connected layer, the output layer predicts the subsequent class based on the input image. Our proposed architecture also follows the same approach where earlier layers consist of convolutional layers. After the convolutional layers, a flattened layer is used, followed by a series of fully connected layers. In the end, the sigmoid function is used to predict the likelihood of the predicted output. Batch Normalization has been used after certain layers to stabilize the training process, whereas average pooling has decreased the dimensionality of the feature maps over the proceeding layers. The black box diagram of the proposed architecture can be seen in Figure 1.

The proposed architecture reads input images with a height and width of 160 pixels each and a batch size of 64. Then the various data augmentation techniques, such as rescaling the input array, rotating the input image randomly between 0 to 360 degrees, horizontal and vertical flip, shear range, and a zoom range of 0.2, are all applied using Keras preprocessing library.

Thus, the proposed architecture accepts input images of size (160,160,3) with all the data augmentation techniques applied. The flow diagram of the proposed architecture can be seen in Figure 2. For the input, at the first layer, 2D convolution operations are performed using filter sizes of (3,3) and 8 different filters. Leaky ReLU is also used at this layer as an activation function. Since it is the first layer extracting image features, it is going to be a high-level feature of input images, and thus, the filter size is kept to be small, i.e., (3,3) instead of a larger filter such as (5,5) or (7,7). With this, we now have the initial feature maps extracted from the input images, but the distributions of input batches can vary a lot for different batches based on the types of images that are included in them. Therefore, it can create problems with the optimizer algorithm's convergence, destabilizing the training process. Thus it is helpful when the input to each layer is unit gaussians. And to do that, these feature maps are batch normalized, which results in a speeded-up training process (faster convergence) and decreased dependency on the weight initialization.

The batch normalized tensor of size (160,160,80) is then passed on to the next block where two convolutional layers are followed, which performs convolution operations with a filter size of (3,3) and 16 different filters each, with Leaky ReLU as activation. It allows us to extract deeper features that could be more meaningful in detecting deepfake images. With these extracted feature maps, they are once again batch normalized. Generally, with deeper CNNs, a larger number of filters will be used in deeper layers to extract deep features. But due to this, the dimensions of the feature maps will keep increasing, resulting in many computations needed as we

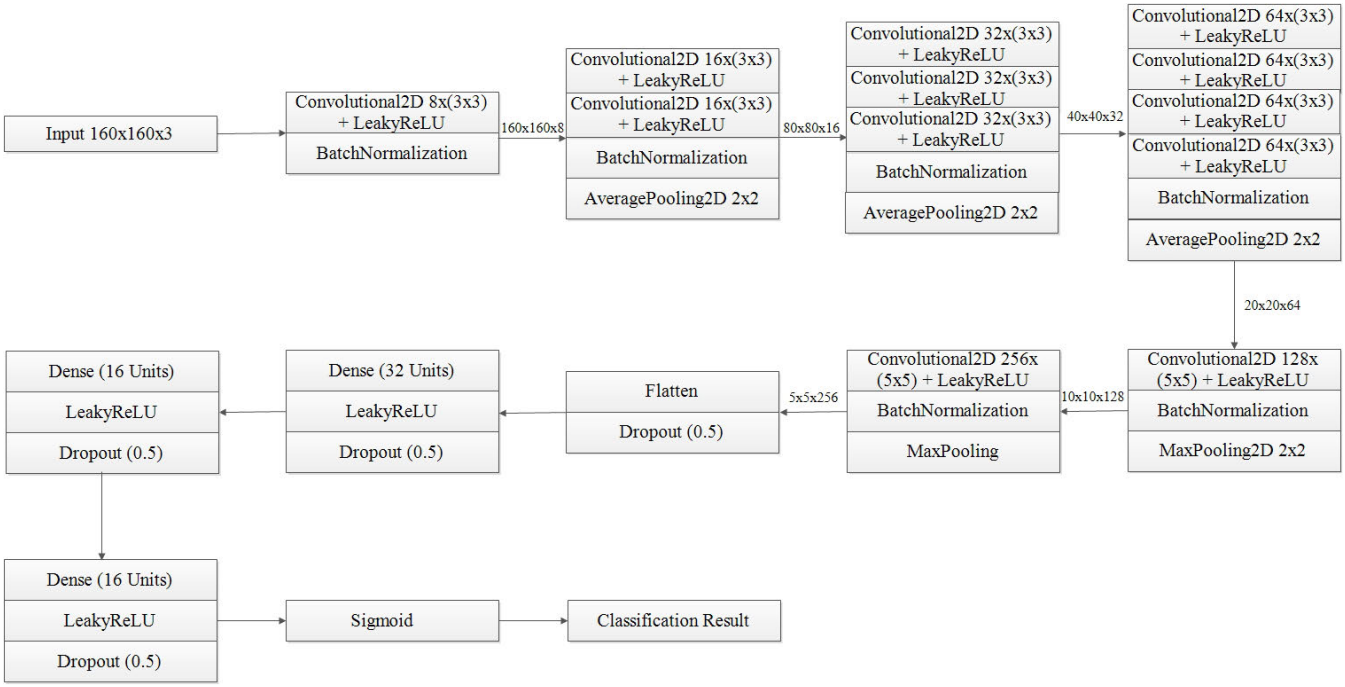

FIGURE 2. Flow diagram of the proposed model.

proceed further. To tackle this issue, pooling layers are used to decrease the dimensionality of the extracted feature maps. With this goal in mind, we have used the average pooling layer of size (2,2), which essentially decreases the dimensions of feature maps by half.

The output from the previous block with Average Pooling layers will be of size (80,80,16). This is accepted as input for the next block, which has a similar structure to the previous block. It differs only by having three convolutional layers with a filter size of (3,3) and 32 different filters, with Leaky ReLU as activation. And then again, batch normalization and average pooling layers are followed. With this pooling layer from the previous block, the dimension of the feature map becomes (40,40,32). And then, it is taken as input for the next block, which consists of 4 consecutive convolutional layers with filter sizes (3,3) and 64 different filters, with Leaky ReLU as activation. It is again followed by batch normalization and average pooling layer.

With this, the next block receives an input size (20, 20, 64). Because of the previous four blocks, we have extracted deep image features, which could be used to classify images as a deepfake or not. So for the next two blocks, we try to use a large filter of size (5,5). For the current block, we use a convolutional layer with (5,5) filter and 128 different filters, with LeakyReLU as activation. Then followed by batch normalization and max pooling layer. This reduces the output dimensions to be (10,10,128). The next block accepts the output from the previous block. It is followed by a convolutional layer with (5,5) filter size and 256 different filters, with LeakyReLU as activation. It is again followed by batch normalization and the Max pooling layer, which gives the dimension output (5,5,256).

The output from the previous block is transformed into a one-dimensional array using the flattened layer.Followed by the flatten layer, there is a dropout layer with value of 0.5, which randomly sets half of the input units to zero. It helps our model to avoid overfitting the training data. Being an improvement over MesoNet, the value of dropout layer has not been changed from it's predecessor which experimentally also yields best results in terms of avoiding overfitting. Following the previous block, there is a fully connected layer with 32 neurons/units. It also utilizes LeakyReLU as an activation function. It is then followed by a dropout layer with a value of 0.5. Similarly, there are two consecutive blocks of the fully connected layer with 16 neurons/units with the LeakyReLU activation function, followed by a dropout layer with a value of 0.5. Finally, there is an output layer with a single neuron and a sigmoid activation function. It predicts whether the input image is a deepfake image or not. If the value is less than 0.5, then the predicted output is real; else, it is a deepfake image. The loss function used during training is binary cross-entropy, and the optimizer used is 'Adam' with a learning rate of 0.01. The black box diagram of the architecture can be seen in Figure 1 and Table 2 describes the output dimensions of each layer along with the number of parameters.

V. RESULTS AND DISCUSSION

This section discusses the performance delivered by the proposed architecture and the results achieved.

TABLE 2. Output dimensions and parameters for each layer.

Layer	Layer Type	Output Dimension	No. of Parameters
1	Input (Convolution 2-D)	(160 x 160 x 8)	224
2	Batch Normalization	(160 x 160 x 8)	32
3	Convolution 2-D	(160 x 160 x 16)	1168
4	Convolution 2-D	(160 x 160 x 16)	2320
5	Batch Normalization	(160 x 160 x 16)	64
6	Average Pooling 2-D	(80 x 80 x 16)	0
7	Convolution 2-D	(80 x 80 x 32)	4640
8	Convolution 2-D	(80 x 80 x 32)	9248
9	Convolution 2-D	(80 x 80 x 32)	9248
10	Batch Normalization	(80 x 80 x 32)	128
11	Average Pooling 2-D	(40 x 40 x 32)	0
12	Convolution 2-D	(40 x 40 x 64)	18496
13	Convolution 2-D	(40 x 40 x 64)	36928
14	Convolution 2-D	(40 x 40 x 64)	36928
15	Convolution 2-D	(40 x 40 x 64)	36928
16	Batch Normalization	(40 x 40 x 64)	256
17	Average Pooling 2-D	(20 x 20 x 64)	0
18	Convolution 2-D	(20 x 20 x 128)	204928
19	Batch Normalization	(20 x 20 x 128)	512
20	Max Pooling 2-D	(10 x 10 x 128)	0
21	Convolution 2-D	(10 x 10 x 256)	819456
22	Batch Normalization	(1 x 10 x 256)	1024
23	Max Pooling 2-D	(5 x 5 x 256)	0
24	Flatten	(6,400)	0
25	Dropout	(6,400)	0
26	Dense	(32)	204832
27	Leaky ReLU	(32)	0
28	Dropout	(32)	0
29	Dense	(16)	528
30	Leaky ReLU	(16)	0
31	Dropout	(16)	0
32	Dense	(16)	272
33	Leaky ReLU	(16)	0
34	Dropout	(16)	0
35	Dense	(1)	17

Total Parameters	1,388,177
Trainable Parameters	1,387,169
Non-Traininable Parameters	1,008

A. SIMULATION SETUP

The Google collab pro has been used for training, which usually assigns Tesla T4 or Tesla P100 GPU. Since Google collab restricts the prolonged usage of GPUs, checkpointing has been used during the training to save the best-performing model based on the lowest validation loss value. If necessary, the training could be resumed from the last best model saved, but it has never been used.

B. DATASET DESCRIPTION

The dataset we decided to use was part of Deepfake Images Detection and Reconstruction Challenge [35]. The dataset consisted of real images from image datasets of CelebA and FFHQ. Both contain 5000 images each. Whereas 1000 images each from GDWCT, AttGAN, STARGAN, StyleGAN and StyleGAN2 datasets are included for deepfake detection. Since the image provided are taken from different types of GAN architecture and datasets, images from these different sources had different resolutions ranging from 1024 × 1024 being the largest to 178 × 218 being the smallest. The resolutions of images are discussed in Table 3.

TABLE 3. Resolution of images from each Data Source.

Type of Image	Dataset	Resolution	No. of Images
Deepfake	GDWCT	216 x 216	1000
Deepfake	AttGAN	256 x 256	1000
Deepfake	StarGAN	245 x 256	1000
Deepfake	StyleGAN	1024 x 1024	1000
Deepfake	StyleGAN2	1024 x 1024	1000
Real	CelebA	178 x 218	5000
Real	FFHQ	1024 x 1024	5000

Thus, there were 10000 real images and 5000 deepfake images. To make a balanced set, we decided to use 5000 real

images only. Thus, to make it completely balanced, we randomly sampled 2500 images from CelebA and 2500 from FFHQ. Thus, a total of 5000 randomly sampled real images from these two sources, whereas we have taken 5000 deepfake images.

We divided the image dataset into a train, validation, and test sets. 60% of images are used for training, 10% for validation, and 30% for test sets (the images have been properly balanced). Firstly, 70% of random sample images are for training from the real dataset from both data sources. We randomly sampled 1750 images from CelebA and 1750 from FFHQ. It makes 3500 real images for training. Out of 3500, we selected every 10th image from this training dataset to be kept as reserved for the validation set. It ensured the ratio of real images from both data sources. It thus gave 350 real images for the validation set.

We then follow the same strategy with deepfakes as well. We sampled 70% of each type of GAN image. That means we sampled 700 images from GDWCT, AttGAN, STARGAN, StyleGAN and StyleGAN2 each. Hence giving a well-balanced set of 3500 deepfake images. And similarly, we selected every 10^{th} image from the training set to be used as a validation set giving 350 deepfake images for the validation set. Thus, now we have 3150 real images and 3150 deepfake images for training, along with 350 real and 350 deepfake images for validation. And the remaining 30% images were used for testing the model for performance after training. For the training purpose, Data Augmentation has been applied to these images. These data augmentation includes vertical flipping, horizontal flipping, zooming by 0.2, shear range by 0.2, width shift range and height shift range by 0.2, as well as random rotation of 360 degrees. These will help the model to learn detect deepfake images while maintaining spatial and scale invariance properties. Since training images consisted of only upright faces that too positioned at the center of the image, there was a very high possibility that the D-CNN model would learn to discriminate between DF and real images based on the features of the center of the images only, that too with upright faces only. To ensure the dataset consisted facial images from different angles, facial images with different spatial position within images and of different scale; data augmentation techniques were used. It helps model to learn spatial and scale invariant features which are of utmost importance for a DF detection system in the wild. The input image size was set to 160×160. Historically detecting low resolution and low quality deepfake images has been considered a difficult task since there is much less information to work with. Adding more to that, conventional social media sites downscale high resolution images to avoid transmission and storage costs. Hence, CNN network with input size of (160,160,3) is selected with the hope to ensure usefulness of the model in real world use case. Low resolution images also helps to keep the computational costs to minimum. But there is scope for future work by either moving to variable sized input NN with Global Pooling layers or experimenting with various efficient upscaling techniques to see performance improvement.

C. TRAINING

During training, the Adam optimizer is used with a learning rate of 0.01. The number of epochs used is 550. Due to the limitations of hardware and time usage on Google Colab, check pointing and CSVLogger have been used to note the training accuracy and training loss as well as validation accuracy and validation loss during the training phase. The batch size is set to 64, stabilising the training phase quite a lot. We save the entire model instead of weights only.

From Figure 3, we can see that the training accuracy steadily increases till the 200^{th} epoch. After the 200^{th} epoch, change in accuracy slowly plateaus over preceding epochs. The same can be seen with loss values during the training phase. Even though not a huge change, performance slowly increases over the epochs. The same trend is also seen in the validation accuracy and loss values in Figure 4. There can be seen fluctuations in the validation accuracy and loss; the most probable reason is the usage of Batch Normalization along with the Dropout layer, aggravating the situation even more. But, apart from those fluctuations, it can be seen in Figure 4 that validation accuracy and validation loss closely follow the same trend as that of Training accuracy and Training loss values. Although validation accuracy and validation loss values fluctuate slightly, it closely follows the training loss value, indicating no overfitting in the model. It can be observed from Figure 5 that there is not much numerical difference between training loss and validation loss which is a good indication for a generalized model and not an overfitting model.

D. EVALUATION METRICS

The proposed model yields 97.2% of accuracy on the Test dataset, which consists of 1500 real and 1500 deepfake images with all the data augmentations techniques applied. As we have already seen during the training phase on the validation set, the model's performance is relatively good, and there are no signs of overfitting in the model. In addition, the Testing dataset's accuracy proves that the model has been trained properly and not overfitted to the training dataset. The accuracy of training, validation and the testing dataset was around 97%. Along with accuracy, the precision, recall, and $F1$ score values are also used to evaluate the model's performance. Along with this, we present the confusion matrix to understand the classification capabilities of the model.

- *Precision*: It refers to the ability of the model to classify positive out of all positive predictions made correctly. It is a metric that indicates how many images were truly deepfake images out of all the images predicted as deepfake by the proposed model. The truly deepfake images classified as deepfake will be considered True Positive. In contrast, those predicted as deepfake but truly were real images will be classified as false positives. The precision formula will be when the true

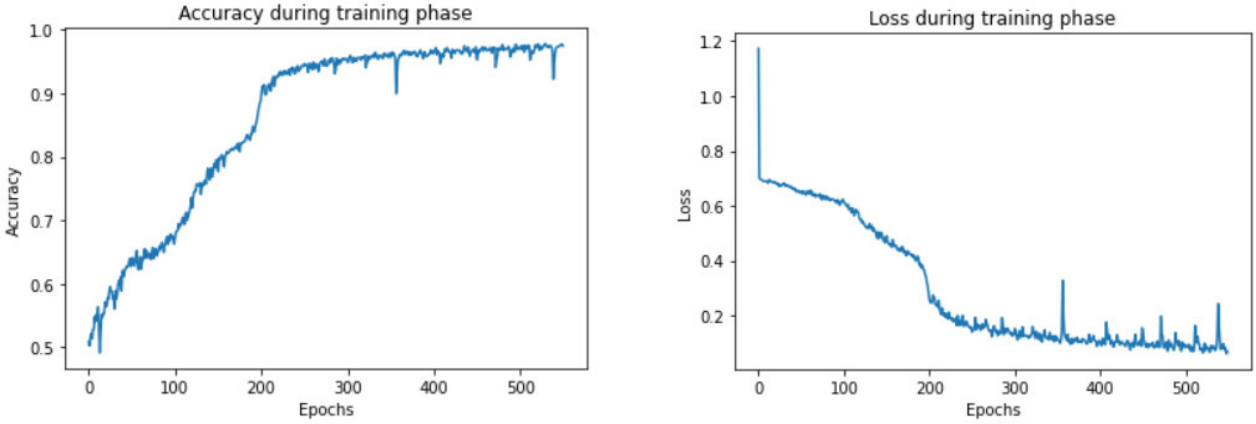

FIGURE 3. Training accuracy and Training Loss over the training epochs.

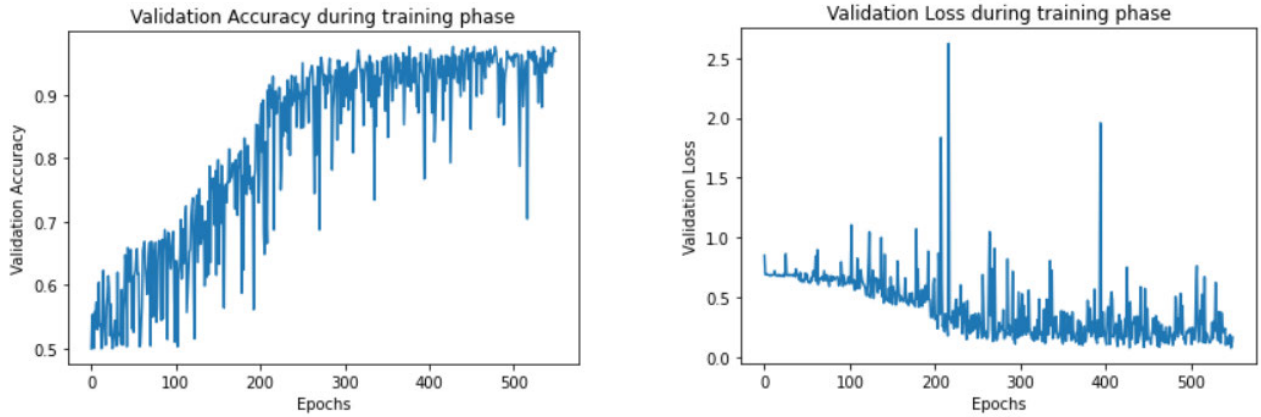

FIGURE 4. Validation accuracy and Validation Loss over the training epochs.

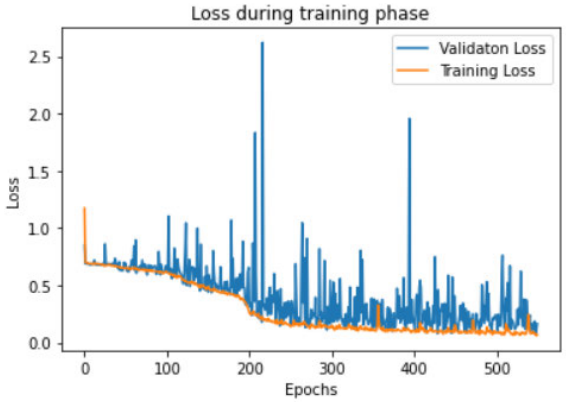

FIGURE 5. Validation accuracy and loss values over the training epochs.

positive is indicated as TP and false positive as FP.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

The precision that we get for our model on the test set is 0.97 for Real images whereas precision for Deepfake images is 0.98.

- *Recall* refers to the model's ability to classify positive positives correctly. So it is a metric which indicates how many images were classified as deepfake out of all the truly deepfake images submitted to the model. So the images that were truly deepfake and classified as deepfake will be considered True Positive, and those which were truly deepfake but misclassified as real images will be considered false negative. The recall formula is when TP indicates True Positives and FN indicates false negatives.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

The recall we get for our model on the test set is 0.98 for Real images, whereas the precision for Deepfake images is 0.97.

- *F1 score*: It indicates the balance between precision and recall. It is the harmonic mean of precision and recall of the proposed approach. It takes into consideration false positives and False Negatives both into consideration. The F1 score is calculated as follows:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

The F1 score for both classes is 0.97, indicating a good balance between precision and recall values.

TABLE 4. Classification report.

	Precision	Recall	f1-score	Support
Real	0.97	0.98	0.97	1500
Deepfake	0.98	0.97	0.97	1500
Macro Average	0.97	0.97	0.97	3000
Weighted Average	0.97	0.97	0.97	3000

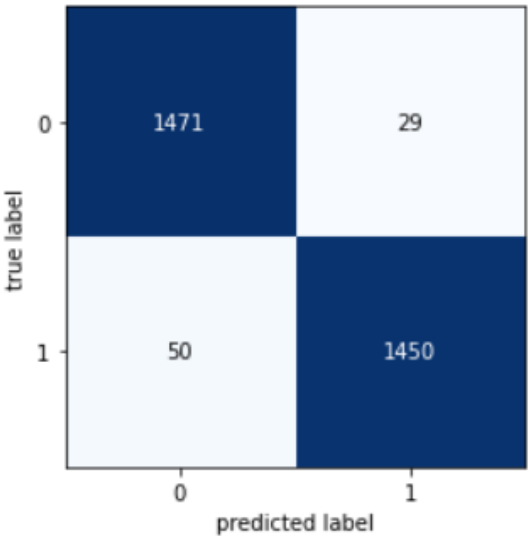

FIGURE 6. Confusion matrix.

The classification report of the performance of our model over the test dataset can be seen in Table 4. Precision, recall, f1-score, macro average, and weighted average can be seen. From all these results, the model's precision, recall, and f1 score show promising results. To further understand the classification capabilities of the proposed model, we have also generated a confusion matrix, shown in Figure 6. Here '0' indicates real, and '1' indicates a deepfake label. True label, which can be seen in the figure, means the real label assigned to it, whereas predicted label means the predicted label from our model. These create 4 categories: true positive, true negative, false positive, and false negative. So category with '0' as a true label and '0' as the predicted label will be considered a true negative since both the labels (true as well as predicted labels) suggest it to be a real image. So they have been classified as negative for deepfake images, which is true. Similarly, the category with true label and predicted label as '1' are true positive. The category with true label as '0' and predicted label as '1' are considered false positives since they were real images misclassified as deepfake by our proposed model. Similarly, the category with true label as '1' and the predicted label as '0' is considered a false negative. Since they were truly deepfake images but were predicted as real. In simpler terms, the confusion matrix shows that out of 1500 Real test images, our model has classified 1471 images correctly, whereas 29 Real images were misclassified as Deepfake. And out of 1500 Deepfake images, our model classifies 1450 images correctly and misclassifies 50 images as Real. Figure 6.

TABLE 5. Performance of the proposed model on individual data source.

Subset of Test set	Proposed Model
AttGAN images + Real images	98.33%
GDWCT images + Real images	99.33%
StyleGAN images + Real images	95.33%
StyleGAN2 + Real images	94.67%
StarGAN + Real images	99.17%

VI. DISCUSSION

To further understand the proposed model's performance. We extend our analysis by evaluating our model over images of all these data sources separately. It will allow us to understand more about the generalizability capabilities of the proposed model. So, in the test set, we had 1500 deepfake images from 5 GAN architectures. Thus, it means we had 300 deepfake images from each data source. When then combined these images from different data sources with real images separately. So we randomly sampled 300 real images, 150 from CelebA and 150 from FFHQ. We then evaluate each model individually. And our model yielded 98.83% accuracy on AttGAN vs CelebA+FFHQ images. Whereas it gave 99.33% accuracy on GDWCT vs CelebA+FFHQ images. It gave 95.33% accuracy on StyleGAN vs CelebA+FFHQ and 94.67% on StyleGAN2 vs CelebA+FFHQ images. Finally, our model yielded 99.17% accuracy on StarGAN vs CelebA+FFHQ images.

We then evaluated the proposed model on the imbalanced set. We already had 300 images for each data source stored separately. We fed all 300 deepfake images separately for each data source to see our model's performance. Our model gave complete 100% accuracy in classifying deepfake images generated from AttGAN with a loss value of 0.0051, whereas on GDWCT, it gave an accuracy of 99.33% with a loss value of 0.0141. Our model performed well over StyleGAN and StyleGAN2 with an accuracy of 95.66% and 93.99%. In contrast, our model gave 99.33% accuracy in classifying images generated using StarGAN. Table 5 presents the performance of the proposed model under different image databases with real images.

The model indicates promising results over the reserved Test set images. The model's performance is balanced over all the different image data sources. When we evaluated our model over all the different data sources separately, we got more insight into the model's performance. It is essential to understand that the accuracy of the combined data set might look promising, but the model might lack performance over certain kinds of images. When we look into it that way, it is seen that our model shows extraordinary performance over the images from AttGAN, GDWCT, and StarGAN. In contrast, performance drops a bit over images from StyleGAN and StyleGAN2.

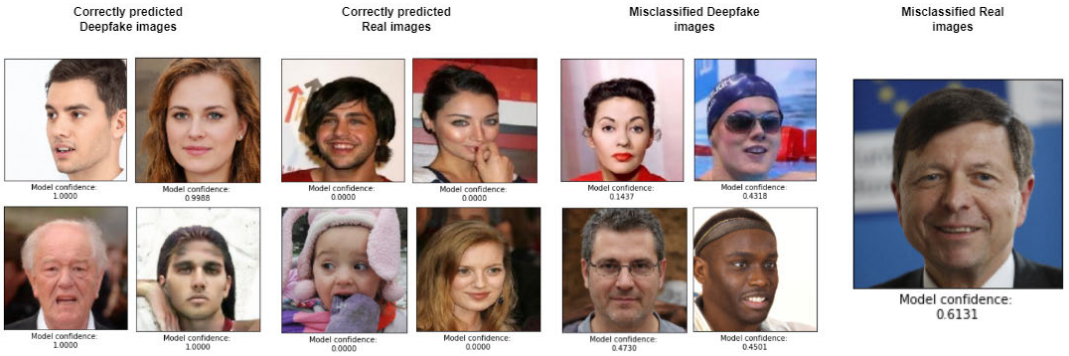

FIGURE 7. Classification of deepfake, real, and misclassified images by the proposed model.

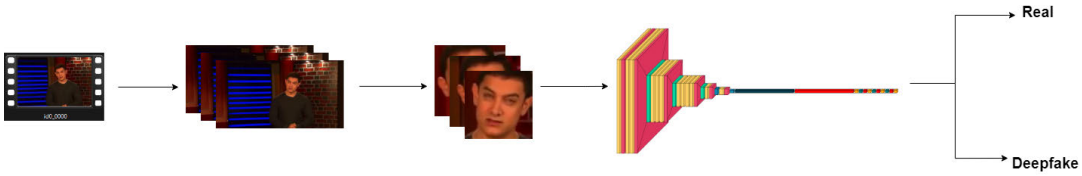

FIGURE 8. Experiment setup for comparing the performance capabilities of the proposed and the existing models.

When investigated further, it is found that StyleGAN and StyleGAN2 images are very high-resolution images, whereas images from AttGAN, GDWCT, and StarGAN are low-resolution images. Thus, it suggests that our model performs extraordinarily over low-resolution images but drops a bit (not much) over high-resolution images. Although still, the performance is quite promising and impressive, even for high-resolution images. But the overall performance, considering the images with such different data sources and resolutions, is still pretty impressive. Some of the results are shown in Figure 7. As it can be seen in the figure, model outputs it's results in terms of confidence score which essentially is probability of that image being a deepfake image or not. If the model confidence score is closer to '0', it is extremely confident about the image being real and vice versa. When the confidence score comes closer to '0.5', it indicates that the model is bit confused. And it can be seen in the figure for misclassified Deepfake images and misclassified Real images, the confidence score is closer to '0.5'. Initial analysis has suggested that since there are manipulation traces and little blurriness left behind for deepfake images, the neurons activation suggests that background areas are activated strongly than facial features such as eyes, nose and mouth. This gets inverted completely for real images where eyes, nose and mouth areas are strongly activated. It suggests that eyes, nose and mouth are far detailed in real images and this becomes the basis for discriminating capabilities of proposed system.

VII. EXPERIMENT

To compare the performance and generalizability capabilities of the proposed model with existing models, we perform a small experiment where we test our proposed model with MesoNet and MesoInception network over the CelebDF dataset. In literature, it is considered a challenging dataset for deepfake detection. Since none of the models are trained over this dataset, it will be an ideal condition to test the generalizability capabilities of these three models. Figure 8 shows the experimental setup of the proposed model.

TABLE 6. Experiment results.

Model	Accuracy
MesoNet	57%
MesoInception	50.73%
Proposed model	77%

CelebDF dataset consists of 795 deepfake videos and 408 real videos. Real videos are divided into 158 real videos provided by authors of celebrities and 250 YouTube videos. We decided to work on 795 deepfake videos and 158 real videos. To simulate deepfake detection in the wild, for both the real and deepfake videos, we extracted every 50th frame of all the videos. We performed face recognition using the Haar Cascade algorithm. Haar cascade was selected for its excellent capabilities of identifying faces irrespective of scale and location within the image. These recognized faces were cropped and stored. Hence, this resulted in a total of 4877 facial images, out of which 3816 were deepfake images and 1061 were real images.

We also implemented MesoNet and MesoInception Network within our local system. We imported pretrained weights provided by the authors. The results for MesoNet, MesoInception network, and the proposed model are 57%, 50.73%, and 77%, respectively. MesoNet delivers 89% accuracy on its native test set of Face2Face images, whereas

MesoInception delivers accuracy of 91%, drops to 57% and 50.73%, respectively. Table 6 shows the experimental results of the proposed model with the existing models. It reflects how achieving generalizability capabilities is an arduous task but, at the same time, of utmost importance for a real-world use case. This drop in accuracy can also be seen in the proposed model, but it still manages to hold its ground. There still lies a scope for future works, which will be discussed in the next section.

VIII. FUTURE SCOPE

As already discussed above, for real-world use of efficient deepfake detection methods, it must be robust, generalizable, computation efficient, and quick. Moreover, there is always a trade-off between accuracy and time latency. As for the proposed model, the future direction could involve experimentation with variable input NN along with Global Pooling so that the resolutions of the input images are not downscaled. Furthermore, experimentation with various efficient image upscaling algorithms and their effects on the performance could also be analyzed for better insights.

IX. CONCLUSION

It has always been challenging to detect deepfake content, as they are generated at a different level of abstraction. It has always been treated as a binary classification problem, as real or deepfake class labels. So, CNN is a prominent solution to detect deepfake images. Motivated by this, we have proposed a CNN-based architecture to detect deepfake images in this paper. The proposed architecture offers 97.2% accuracy considering images from 5 different data sources for deepfake images and 2 different data sources for real images. Even though there is a huge difference between the resolutions of these images, the proposed architecture provides a well-balanced performance over all data sources. The work can be further extended to classify video deepfake content. This model can be used for video deepfake detection, where each video frame is extracted, the face is detected, cropped, and then fed to the model to identify deepfake manipulations. This can be easily done by creating a pipeline to process this video data. Thus, the proposed CNN-based model performs well and has quite a balanced performance over the given dataset with all the data augmentation techniques applied. Furthermore, it shows good generalizability and performance over unseen reserved test sets.

REFERENCES

[1] I. Perov, D. Gao, N. Chervoniy, K. Liu, S. Marangonda, C. Umé, M. Dpfks, C. S. Facenheim, R. P. Luis, J. Jiang, and S. Zhang, "DeepFaceLab: Integrated, flexible and extensible face-swapping framework," 2020, *arXiv:2005.05535*.

[2] K. N. Ramadhani and R. Munir, "A comparative study of deepfake video detection method," in *Proc. 3rd Int. Conf. Inf. Commun. Technol. (ICOIACT)*, 2020, pp. 394–399.

[3] R. Katarya and A. Lal, "A study on combating emerging threat of deepfake weaponization," in *Proc. 4th Int. Conf. I-SMAC*, 2020, pp. 485–490.

[4] D. Yadav and S. Salmani, "Deepfake: A survey on facial forgery technique using generative adversarial network," in *Proc. Int. Conf. Intell. Comput. Control Syst. (ICCS)*, May 2019, pp. 852–857.

[5] 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. 18757–18775, 2022.

[6] C. C. K. Chan, V. Kumar, S. Delaney, and M. Gochoo, "Combating deepfakes: Multi-LSTM and blockchain as proof of authenticity for digital media," in *Proc. IEEE/ITU Int. Conf. Artif. Intell. Good (AI4G)*, Sep. 2020, pp. 55–62.

[7] *DeepFaceLab*. Accessed: Jan. 14, 2022. [Online]. Available: https://github.com/iperov/DeepFaceLab

[8] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2Face: Real-time face capture and reenactment of RGB videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2387–2395.

[9] P. Korshunov and S. Marcel, "Vulnerability assessment and detection of deepfake videos," in *Proc. Int. Conf. Biometrics (ICB)*, Jun. 2019, pp. 1–6.

[10] Y. Mirsky and W. Lee, "The creation and detection of deepfakes," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–7, 2021.

[11] *A Video That Appeared to Show Obama Calling Trump a 'Dipsh-T' is a Warning About a Disturbing New Trend Called 'Deepfakes'*. Accessed: May 25, 2022. [Online]. Available: https://www.businessinsider.in/tech/a-video-that-appeared-to-show-obama-calling-trump-a-dipsh-t-is-a-warning-about-a-disturbing-new-trend-called-deepfakes/articleshow/63807263.cms

[12] P. Yu, Z. Xia, J. Fei, and Y. Lu, "A survey on deepfake video detection," *IET Biometrics*, vol. 10, no. 6, pp. 607–624, Nov. 2021.

[13] Y.-J. Heo, Y.-J. Choi, Y.-W. Lee, and B.-G. Kim, "Deepfake detection scheme based on vision transformer and distillation," 2021, *arXiv:2104.01353*.

[14] R. Caldelli, L. Galteri, I. Amerini, and A. D. Bimbo, "Optical flow based CNN for detection of unlearnt deepfake manipulations," *Pattern Recognit. Lett.*, vol. 146, pp. 31–37, Jun. 2021.

[15] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2242–2251.

[16] K. Patel, D. Mehta, C. Mistry, R. Gupta, S. Tanwar, N. Kumar, and M. Alazab, "Facial sentiment analysis using AI techniques: State-of-the-art, taxonomies, and challenges," *IEEE Access*, vol. 8, pp. 90495–90519, 2020.

[17] H. S. Shad, M. M. Rizvee, N. T. Roza, S. M. A. Hoq, M. M. Khan, A. Singh, A. Zaguia, and S. Bourouis, "Comparative analysis of deepfake image detection method using convolutional neural network," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–18, Dec. 2021.

[18] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The deepfake detection challenge (DFDC) dataset," 2020, *arXiv:2006.07397*.

[19] J. Hathaliya, R. Parekh, N. Patel, R. Gupta, S. Tanwar, F. Alqahtani, M. Elghatwary, O. Ivanov, M. S. Raboaca, and B.-C. Neagu, "Convolutional neural network-based Parkinson disease classification using SPECT imaging data," *Mathematics*, vol. 10, no. 15, p. 2566, Jul. 2022.

[20] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: A compact facial video forgery detection network," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2018, pp. 1–7.

[21] 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)*, Jun. 2020, pp. 3204–3213.

[22] K. Chugh, P. Gupta, A. Dhall, and R. Subramanian, "Not made for each other- audio-visual dissonance-based deepfake detection and localization," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 439–447.

[23] Y. Zhang, J. Zhan, W. Jiang, and Z. Fan, "Deepfake detection based on incompatibility between multiple modes," in *Proc. Int. Conf. Intell. Technol. Embedded Syst. (ICITES)*, Oct. 2021, pp. 1–7.

[24] Y. Li, M.-C. Chang, and S. Lyu, "In ictu oculi: Exposing AI created fake videos by detecting eye blinking," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2018, pp. 1–7.

[25] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, "Detection of GAN-generated fake images over social networks," in *Proc. IEEE Conf. Multimedia Inf. Process. Retr. (MIPR)*, Apr. 2018, pp. 384–389.

[26] C.-C. Hsu, C.-Y. Lee, and Y.-X. Zhuang, "Learning to detect fake face images in the wild," in *Proc. Int. Symp. Comput., Consum. Control (IS3C)*, Dec. 2018, pp. 388–391.

[27] D. Guera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.

[28] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 8261–8265.
[29] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose deepfakes and face manipulations," in *Proc. IEEE Winter Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2019, pp. 83–92.
[30] F. F. Kharbat, T. Elamsy, A. Mahmoud, and R. Abdullah, "Image feature detectors for deepfake video detection," in *Proc. IEEE/ACS 16th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2019, pp. 1–4.
[31] Z. Akhtar and D. Dasgupta, "A comparative evaluation of local feature descriptors for DeepFakes detection," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Nov. 2019, pp. 1–5.
[32] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 2307–2311.
[33] I. Amerini, L. Galteri, R. Caldelli, and A. D. Bimbo, "Deepfake video detection through optical flow based CNN," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 1205–1207.
[34] M. Dordevic, M. Milivojevic, and A. Gavrovska, "DeepFake video analysis using SIFT features," in *Proc. 27th Telecommun. Forum (TELFOR)*, Nov. 2019, pp. 1–4.
[35] *Deepfake Images Detection and Reconstruction Challenge—21st International Conference on Image Analysis and Processing*. Accessed: Jan. 5, 2023. [Online]. Available: https://iplab.dmi.unict.it/Deepfake challenge/

YOGESH PATEL received the M.Tech. degree in computer engineering from the Institute of Technology, Nirma University. He is currently working on presenting solutions to integrate generative adversarial networks in adversarial learning techniques in a wide range of domains, such as healthcare, vehicular networks, and emerging communication networks. His research interests include deep learning, data science, and blockchain.

SUDEEP TANWAR (Senior Member, IEEE) is currently a Professor with the Computer Science and Engineering Department, Institute of Technology, Nirma University, India. He is also a Visiting Professor with Jan Wyzykowski University, Polkowice, Poland, and the University of Pitesti, Pitesti, Romania. He has authored two books, edited 13 books, and published more than 270 technical papers in top journals and conferences, such as IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE WIRELESS COMMUNICATIONS, IEEE NETWORK, ICC, GLOBECOM, and INFOCOM. His H-index is 58. He initiated research in the field of blockchain technology adoption in various verticals, in 2017. He actively serves his research communities in various roles. His research interests include blockchain technology, wireless sensor networks, fog computing, smart grids, and the IoT. He is a member of the Technical Committee on Tactile Internet of the IEEE Communication Society and a Senior Member of CSI, IAENG, ISTE, and CSTA. He has received the Best Research Paper Awards from IEEE GLOBECOM 2018, IEEE ICC 2019, and Springer ICRIC-2019. He has served for many international conferences as a member of the organizing committee, such as the Publication Chair for FTNCT-2020, ICCIC 2020, and WiMob2019; a member of the Advisory Board for ICACCT-2021 and ICACI 2020; the Workshop Co-Chair for CIS 2021; and the General Chair for IC4S 2019 and 2020 and ICCSDF 2020. He is serving on the editorial boards for *Frontiers of Blockchain*, *Cyber Security and Applications*, *Computer Communications*, the *International Journal of Communication Systems*, and *Security and Privacy*.

PRONAYA BHATTACHARYA (Member, IEEE) received the Ph.D. degree from Dr. A. P. J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh, India. He is currently an Associate Professor with the Computer Science and Engineering Department, Amity School of Engineering and Technology, Amity University, Kolkata, India. He has over ten years of teaching experience. He has authored or coauthored more than 100 research papers in leading SCI journals and top core IEEE COMSOC A* conferences. Some of his top-notch findings are published in reputed SCI journals, such as IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE ACCESS, IEEE SENSORS JOURNAL, *IEEE Internet of Things Magazine*, *IEEE Communication Standards Magazine*, *ETT* (Wiley), *Expert Systems* (Wiley), *CCPE* (Wiley), *FGCS* (Elsevier), *OQEL* (Springer), *WPC* (Springer), ACM-MOBICOM, IEEE-INFOCOM, IEEE-ICC, IEEE-CITS, IEEE-ICIEM, IEEE-CCCI, and IEEE-ECAI. His H-index is 19 and i10-index is 32. His research interests include healthcare analytics, optical switching and networking, federated learning, blockchain, and the IoT. He is an Active Member of the ST Research Laboratory. He has been a keynote speaker, a technical committee member, and a session chair. He is a recipient of eight Best Paper Awards from Springer ICRIC-2019, IEEE-ICIEM-2021, IEEE-ECAI-2021, Springer COMS2-2021, and IEEE-ICIEM-2022. He is a Reviewer of 21 reputed SCI journals, such as IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS OF VEHICULAR TECHNOLOGY, IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, IEEE ACCESS, *IEEE Network*, *ETT* (Wiley), *IJCS* (Wiley), *MTAP* (Springer), *OSN* (Elsevier), and *WPC* (Springer).

RAJESH GUPTA received the B.E. degree from the University of Jammu, India, in 2008, the master's degree in technology from Shri Mata Vaishno Devi University, Jammu, in 2013, and the Ph.D. degree in computer science and engineering from Nirma University, Ahmedabad, Gujarat, India, in 2023, under the supervision of Dr. Sudeep Tanwar. He is an Assistant Professor with Nirma University. He has authored/coauthored some publications (including papers in SCI indexed journals and IEEE ComSoc sponsored international conferences). Some of his research findings are published in top-cited journals and conferences, such as IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, *IEEE Network magazine*, IEEE INTERNET OF THINGS JOURNAL, *IEEE Internet of Things Magazine*, *Computer Communications*, *Computer and Electrical Engineering*, *International Journal of Communication Systems* (Wiley), *Transactions on Emerging Telecommunications Technologies* (Wiley), *Physical Communication* (Elsevier), IEEE ICC, IEEE INFOCOM, IEEE GLOBECOM, and IEEE CITS. His H-index is 27 and i10-index is 37. His research interests include device-to-device communication, network security, blockchain technology, 5G communication networks, and machine learning. He is an Active Member of the ST Research Laboratory. He was a recipient of the Doctoral Scholarship from the Ministry of Electronics and Information Technology, Government of India, under the Visvesvaraya Ph.D. Scheme, and the Student Travel Grant from WICE-IEEE for attending IEEE ICC 2021, which was held in Canada. He received the Best Research Paper Award from IEEE ECAI 2021, IEEE ICCCA 2021, IEEE IWCMC 2021, and IEEE SCIoT 2022. His name has been included in the list of Top 2% scientists worldwide published by Stanford University, USA, in 2021 and 2022. He was felicitated by Nirma University for his research achievements, in 2019–2020 and 2021–2022. For more information, see www.sudeeptanwar.in.

TURKI ALSUWIAN received the B.Sc. degree from King Saud University, Riyadh, Saudi Arabia, the M.Sc. degree from Gannon University, PA, USA, in 2011, and the Ph.D. degree from the University of Dayton, OH, USA, in 2018, all in electrical engineering. He was an Electrical Power Engineer with Saudi Electricity Company, from April 2004 to January 2009. He is currently an Assistant Professor with the Electrical Engineering Department, Najran University, Saudi Arabia. His main research interests include applied control in different fields, such as flight control, power quality control, power electronics control, communication control, adaptive control, modeling control, and artificial intelligence.

INNOCENT EWEAN DAVIDSON (Senior Member, IEEE) received the B.Sc. (Eng.) (Hons.) and M.Sc. (Eng.) degrees in electrical engineering from the University of Ilorin, in 1984 and 1987, respectively, the Ph.D. degree in electrical engineering from the University of Cape Town, in 1998, and the PG Diploma degree in business management from the University of KwaZulu-Natal, in 2004. He also received the Associate certificate in sustainable energy management (SEMAC) from the British Columbia Institute of Technology, Burnaby, BC, Canada, in 2011, and the Course certificate in artificial intelligence from the University of California at Berkeley, USA, in 2020. He was a Full Professor and the Chair of the Department of Electrical Power Engineering, the Research Leader of the Smart Grid Research Centre, and the Program Manager of the DUT-DSI Space Science and CNS Research Program, Durban University of Technology (DUT), Durban, South Africa, from 2016 to 2022. Currently, he is a Full Professor with the Africa Space Innovation Center (ASIC), French South African Institute of Technology (F'SATI), Cape Peninsula University of Technology (CPUT), Bellville, South Africa. He is also a Senior Lecturer with the Department of Electrical Engineering Technology, University of Johannesburg. He has supervised five postdoctoral research fellows, and graduated 55 Ph.D./master's students and over 1200 engineers, technologists, and technicians. He is the author/coauthor of over 350 technical papers in accredited journals and peer-reviewed conference proceedings. He has managed over U.S. $3 million in research funds. His current research interests include HVdc power transmission, grid integration of renewable energy, applied artificial Intelligence, and space technology. He is a fellow of the Institute of Engineering and Technology, U.K., and the South African Institute of Electrical Engineers; a Chartered Engineer in the U.K.; and a registered Professional Engineer (P.Eng.) of the Engineering Council of South Africa. He is a member of the Western Canada Group of Chartered Engineers (WCGCE), the Institute of Engineering and Technology (IET Canada) British Columbia Chapter, IEEE Collabratec Communities on Smart Cities, and IEEE South Africa Chapter. He was the General Chair of the 30th IEEE Southern Africa Universities Power Engineering Conference, in 2022. He is the Host and Convener of the DSI-DUTSANSA-ATNS Space Science and CNS Symposium, and a Guest Speaker in several forums, including the Science Forum of South Africa and the International Conference on Sustainable Development. He was a recipient of numerous international Best Paper Awards from DUT's Annual Research and Innovation. He is a C2-rated researcher from the National Research Foundation (NRF), South Africa.

THOKOZILE F. MAZIBUKO received the National Diploma degree in electrical engineering from Durban University of Technology (DUT), in 2008, the B.Tech. degree in electrical engineering from Tshwane University of Technology (TUT), in 2011, and the joint master's degree (cum laude) in electrical engineering from TUT and RWTH Aachen University, Germany, in 2014. She is currently pursuing the Ph.D. degree with DUT. She was a Tutor/Student Assistant with DUT, from 2006 to 2007, and an Engineering Trainee with Anglo Platinum, from 2007 to 2008. She was a Planner Assistant and a Quality Control Coordinator with Anglo American/Hatch, Rustenburg, South Africa, from 2008 to 2009. During her master's degree, she conducted research and implementation of real-time platforms, namely, the application of PTP synchronized PMUs in power system small signal stability and transient stability analysis of a multi-machine systems based on synchro-phasors. She was with the Council for Scientific and Industrial Research (CSIR), Pretoria, until 2016, and with Rand Water, Johannesburg, from 2017 to 2018. She was a Lecturer with the University of Johannesburg, from 2018 to 2020. In January 2021, she joined as a Lecturer at DUT. Her research interests include smart micro-grids, network optimization, control, and applied artificial intelligence.