Acadlore Transactions on AI and Machine Learning
https://www.acadlore.com/journals/ATAIML

Enhancing Face Spoofing Attack Detection: Performance Evaluation of a VGG-19 CNN Model

Thomas Ayanwola*, Awodele Oludele, Michael Agbaje

Department of Computer Science, Babcock University, 121003 Ilishan, Nigeria

* Correspondence: Thomas Ayanwola (ayanwola0529@pg.babcock.edu.ng)

Received: 04-14-2023 **Revised:** 05-10-2023 **Accepted:** 05-29-2023

Citation: T. Ayanwola, A. Oludele, and M. Agbaje, "Enhancing face spoofing attack detection: Performance evaluation of a VGG-19 CNN model," *Acadlore Trans. Mach. Learn.*, vol. 2, no. 2, pp. 84–98, 2023. https://doi.org/10.56578/ataiml020204.

Abstract: With the wide use of facial verification and authentication systems, the performance evaluation of Spoofing Attack Detection (SAD) module in the systems is important, because poor performance leads to successful face spoofing attacks. Previous studies on face SAD used a pretrained Visual Geometry Group (VGG) -16 architecture to extract feature maps from face images using the convolutional layers, and trained a face SAD model to classify real and fake face images, obtaining poor performance for unseen face images. Therefore, this study aimed to evaluate the performance of VGG-19 face SAD model. Experimental approach was used to build the model. VGG-19 algorithm was used to extract Red Green Blue (RGB) and deep neural network features from the face datasets. Evaluation results showed that the performance of the VGG-19 face SAD model improved by 6% compared with the state-of-the-art approaches, with the lowest equal error rate (EER) of 0.4%. In addition, the model had strong generalization ability in top-1 accuracy, threshold operation, quality test, fake face test, equal error rate, and overall test standard evaluation metrics.

Keywords: Security; Biometric; Face; Spoofing; VGG-19; Evaluation; Performance

1 Introduction

Biometric user verification and authentication systems have become increasingly popular for applications such as device unlocking, automatic e-transactions, border security, airport control, attendance systems at colleges and universities, face payment, and electronic polling [1]. Among various biometric traits, facial recognition algorithms have gained widespread adoption due to their convenience and contactless nature [2]. These systems work by comparing face images captured by cameras against a database of known face images to identify matches [3].

However, face spoofing attacks (FSAs) pose a significant challenge to the security of facial verification and authentication systems. In FSAs, unauthorized individuals impersonate registered users by obtaining their facial data through social media networks or other means, and then use the acquired data to deceive the system and gain unauthorized access [4, 5]. To ensure robust security in real-world scenarios, face spoofing attack detection (SAD) models must be reliable and demonstrate strong performance on unseen face images [6].

Previous approaches to face SAD mainly relied on extracting Local Binary Patterns (LBP) feature histograms from face patches, such as eyes, nose, and mouth regions, using convolutional layers of fine-tuned Visual Geometry Group (VGG) 11-16 architectures [7]. Despite training various facial parts and three Convolutional Neural Network (CNN) architectures, these models exhibited poor performance and a high rate of successful FSAs [8]. Other studies proposed the use of Discrete Cosine Transform (DCT) and LBP features [9, 10], Histogram of Oriented Gradients (HOG) features [11], and Gabor wavelet features [12]. However, these features lack sufficient distinguishing traits to accurately classify real and fake images, resulting in overfitted face SAD models with poor performance [13].

Existing face SAD models suffer from inadequate training datasets and limited feature extraction capabilities, leading to poor detection of unseen FSAs in real-time scenarios [14, 15]. Although some researchers have explored the application of CNNs for face SAD [16], the potential of the Visual Geometry Group-19 (VGG-19) CNN architecture and joint learning of Red Green Blue (RGB) and deep network features for improved classification has not been fully investigated.

This study aims to evaluate the performance of a VGG-19 face SAD model that leverages joint learning of RGB and deep features to enhance the detection of spoofing attacks. The evaluation results and comparisons with state-of-the-art approaches will be presented in detail, along with an analysis of the model's generalization capabilities across various evaluation metrics.

2 Methodology

The face SAD models, namely, VGG-19A, VGG-19B, VGG-19C, and VGG-19D were implemented using Python programming language, Google Colaboratory, and TensorFlow 2.0 framework [17]. TensorBoard, a visualization tool provided with TensorFlow, was used for tracking loss and accuracy experiment metrics, and visualizing the model graph [18]. Face datasets were used for the end-to-end training of VGG-19 architecture and its three derived network [19]. The primary sources of the face datasets were Nanjing University of Aeronautics and Astronautics (NUAA), Chinese Academy of Sciences' Institute of Automation (CASIA), OUL University (OULU), Wide Multi Channel presentation Attack (WMCA), 3D Mask Attack Dataset (3DMAD), and CASIA-Face-Africa [20]. Then the trained models were tested with unknown face datasets, and evaluated using the metrics in the literature, such as top-1 accuracy, threshold operation accuracy, quality test, fake face test, EER, and overall test [21]. The model with the best evaluation result was used as the face SAD model. Each dataset contained training, validation, and testing face images, which were further divided into real and fake faces [22]. The spoof attacks in the datasets included mask, photo, and video attacks [22]. Table 1 depicts the basic characteristics of each of the six datasets.

The raw face datasets were pre-processed for detecting, cropping and removing dirty face images. Singular Value Decomposition (SVD) tool was used for pre-processing, which detected faces and facial landmarks on images and resized the face images to 244x244x3. The datasets contained 80,000 training set and 16,000 test set after pre-processing [23].

The datasets had totally 85,571 real and fake faces, which were pre-processed into their bit values, and labeled according to RGB pixel using SVD. After removing 5,571 dirty face images, the rest 80,000 real and fake face images remained. 80% of the datasets was the training set and 20% was the test set. The training set had 64,000 face images, with 32,000 real ones and 32,000 fake ones. A subset of training set was used as validation set, with 3,200 real face images and 3,200 fake ones. The test set had 16,000 face images, with 8,000 real ones and 8,000 fake ones.

The input raw face images were pre-processed and visualized before training [24]. The VGG-19A, VGG-19B, VGG-19C, and VGG-19D were trained using RMSProp gradient based optimization algorithm, with an initial learning rate of 0.00002 [25]. The neural networks were trained for a total of 89 to 100 epochs [26]. The TensorFlow optimal learning rate finder was used after 10 and 15 epochs [27].

Considering the size of the face images database, the total time for training model in Google Collaboratory Cloud environment with NVIDIA GeForce RTX 4090 GPU was 16 hours [28]. The face images were extracted from the six datasets to form the face spoofing detection database [29].

Table 1. Basic characteristics of face SAD datasets

S/N	Datasets	Number of subjects	Number of face images	Number of real images	Number of fake images	Modal types	Spoof attack types
1	3DMAD	12	255	100	155	RGB/Depth	Mask
2	CASIA	1000	21000	10500	10500	RGB/Depth/R	Photo, and video
3	NUAA	15	12614	5105	7509	RGB	Photo
4	OULU	55	5940	1980	3960	RGB	Photo, and video
5	WMCA	72	6716	3358	3358	RGB/Depth/ IR/Thermal	Photo, video, and mask
6	CASIA-Face-Africa	1183	38546	19273	19273	RGB	Photo

3 Results

The training and validation results of the VGG-19A model are shown in Table 2 and Figure 1.

Table 2. Training and validation results of the VGG-19A model

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
1/100	0.2858	0.8971	0.5654	0.7672
2/100	0.2240	0.9141	0.5211	0.7988
3/100	0.2058	0.9194	0.4884	0.8100
4/100	0.1944	0.9236	0.5188	0.8064
5/100	0.1868	09249	0.5525	0.8050
6/100	0.1812	09281	0.6030	0.8016
7/100	0.1741	0.9316	0.5959	0.8084
8/100	0.1702	0.9330	0.5497	0.8140
9) 100	0.1656	0.9340	0.6051	0.8100
10/100	0.1609	0.9367	0.5918	0.8118
11/100	0.1565	0.9384	0.6218	0.8104
12/100	0.1520	0.9410	0.5910	0.8182
13/100	0.1477	0.9426	0.5864	0.8152
14100	0.1443	0.9444	0.5927	0.8190
15/100	0.1399	09467	0.6234	0.8182
16/100	0.1365	0.9481	0.6289	0.8176
17/100	0.1317	0.9508	0.5956	0.8240
18/100	0.1285	0.9529	0.6696	0.8190
19/100	0.1258	0.9536	0.6419	0.8246
20/100	0.1218	0.9556	0.7354	0.8206
21/100	0.1173	0.9558	0.6038	0.8296
22/100	0.1138	0.9597	0.8082	0.8110
23/100	0.1100	0.9614	0.6923	0.8226
24/100	0.1065	0.9622	0.8417	0.8096
25/100	0.1031	0.9636	0.8782	0.8106
26/100	0.0995	0.9655	0.7675	0.8228
27/100	0.0965	0.9674	0.7405	0.8272
28/100	0.0924	0.9698	0.7064	0.8308
29/100	0.0898	0.9706	0.8736	0.8150
30/100	0.0849	0.9725	0.8903	0.8116
31/100	0.0822	0.9739	0.8691	0.8182
32/100	0.0788	09764	0.8833	0.8130
33/100	0.0758	0.9772	0.8148	0.8254
34/100	0.0723	0.977	0.8831	0.8248
35/100	0.0692	0.979	0.9101	0.8150
36/100	0.0662	0.9809	0.9683	0.8188
37/100	0.0622	0.9826	0.8967	0.8262
38/100	0.0598	0.9839	1.0424	0.8156
39/100	0.0581	0.9844	0.971	0.8218
40/100	0.0537	0.9859	1.1659	0.8110
41/100	0.0510	0.9874	1.0033	0.8192
42/100	0.0489	0.9879	1.0974	0.8192
43/100	0.0457	0.9888	0.9876	0.8292
44/100	0.0435	0.9902	1.0909	0.8204
45/100	0.0401	0.9915	0.9841	0.8322
46/100	0.0385	0.9914	1.1322	0.8232
47/100	0.0361	0.9918	1.1609	0.8202
48/100	0.0339	0.9931	1.1549	0.8208
49 / 100	0.0319	0.9929	1.1243	0.8284
50 / 100	0.0296	0.9941	1.1067	0.8276
51 / 100	0.0265	0.9949	1.4574	0.8112
52 / 100	0.0253	0.9948	1.2037	0.8274
53 / 100	0.0237	0.9959	1.3284	0.8202
54 / 100	0.0215	0.9961	1.2718	0.8276

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
55 / 100	0.0199	0.9966	1.3732	0.8202
56 / 100	0.0187	0.9972	1.4028	0.8210
57 / 100	0.0166	0.9979	1.2637	0.8262
58 / 100	0.0152	0.9981	1.3402	0.8216
59 / 100	0.0139	0.9981	1.4616	0.8240
60 / 100	0.0128	0.9988	1.5826	0.8158
61 / 100	0.0115	0.9986	1.4762	0.8256
62 / 100	0.0107	0.9990	1.3859	0.8314
63 / 100	0.0096	0.9989	1.5240	0.8270
64 / 100	0.0086	0.9991	1.4305	0.8292
65 / 100	0.0078	0.9995	1.6003	0.8260
66 / 100	0.0067	0.9997	1.7436	0.8234
67 / 100	0.0061	0.9994	1.5660	0.8238
68 / 100	0.0053	0.9995	1.7274	0.8258
69 / 100	0.0050	0.9996	1.6954	0.8240
70 / 100	0.0042	0.9997	1.6457	0.8290
71 / 100	0.0036	0.9997	1.9800	0.8188
72 / 100	0.0033	0.9997	1.9834	0.8156
73 / 100	0.0028	0.9998	1.7679	0.8282
74 / 100	0.0024	0.9997	1.9357	0.8208
75 / 100	0.0022	0.9999	2.0402	0.8268
76 / 100	0.0016	0.9999	2.1737	0.8236
77 / 100	0.0017	0.9998	2.1998	0.8240
78 / 100	0.0013	0.9999	1.9325	0.8294
79 / 100	0.0012	0.9999	2.2198	0.8266
80 / 100	0.0011	0.9998	2.0579	0.8252
81 / 100	8.35E-04	0.9999	2.2029	0.8280
82 / 100	8.06 E-04	0.9998	2.1431	0.8300
83 / 100	4.82E-04	0.9999	2.1699	0.8304
84 / 100	3.99E-04	1	2.5538	0.8244
85 / 100	421 E-04	1	2.3126	0.8284
86 / 100	5.41E-04	0.9999	2.2783	0.8310
87 / 100	2.51 E-04	1	2.5194	0.8276
88 / 100	2.86E-04	1	2.6917	0.8232
89 / 100	2.51 E-04	1	2.6306	0.8268

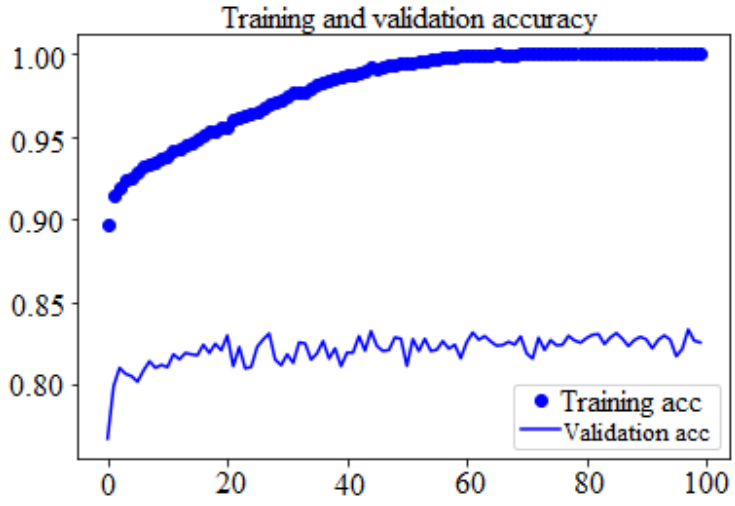

Figure 1. Training and validation accuracy of the VGG-19A model

The training and validation results of the VGG-19B model are shown in Table 3 and Figure 2.

Table 3. Training and validation results of the VGG-19B model

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
1/100	0.3138	0.8882	0.5966	0.7788
2/100	0.244	0.9079	0.4973	0.8082
3/100	0.2229	0.914	0.5208	0.8082
4/100	0.2122	0.918	0.5499	0.8086
5/100	0.2029	0.9195	0.5629	0.8056
6/100	0.1977	0.9221	0.5736	0.8054
7/100	0.1911	0.9245	0.6038	0.8018
8/100	0.185	0.9272	0.5207	0.8186
9/100	0.1817	0.927	0.5646	0.8154
10/100	0.1788	0.9301	0.612	0.8150
11/100	0.1749	0.9316	0.6354	0.8132
12/100	0.1731	0.9317	0.5743	0.8200
13/100	0.1676	0.9329	0.5841	0.8196
14/100	0.1641	0.9381	0.6317	0.8176
15/100	0.1628	0.9361	0.6492	0.8176
16/100	0.1583	0.9403	0.6632	0.8188
17/100	0.1554	0.9419	0.5346	0.8294
18/100	0.1523	0.9413	0.7164	0.8118
19/100	0.1491	0.943	0.6535	0.8206
20/100	0.1468	0.9442	0.7222	0.8170
21/100	0.1436	0.944	0.6846	0.8158
22/100	0.141	0.9451	0.6906	0.8218
23/100	0.1384	0.9456	0.735	0.8198
24/100	0.1318	0.9508	0.7863	0.8146
25/100	0.1329	0.9523	0.7608	0.8206
26/100	0.1294	0.9523	0.7852	0.8210
27/100	0.1256	0.9555	0.7489	0.8228
28/100	0.1241	0.9547	0.7982	0.8166
29/100	0.1197	0.9565	0.7759	0.8250
30/100	0.1159	0.9587	0.7712	0.8218
31/100	0.1154	0.9584	0.6788	0.8320
32/100	0.1108	0.9618	0.78	0.8270
33/100	0.1077	0.9624	0.7895	0.8286
34/100	0.1052	0.9644	0.7135	0.8322
35/100	0.1022	0.9644	0.7716	0.8274
36/100	0.0979	0.9686	0.7806	0.8300
37/100	0.0963	0.9678	0.9526	0.8140
38/100	0.0949	0.969	0.8005	0.8284
39/100	0.0917	0.9689	0.822	0.8296
40/100	0.0865	0.9729	0.8628	0.8304
41/100	0.0823	0.9738	0.8923	0.8286
42/100	0.0812	0.9755	0.9835	0.8226
43/100	0.0796	0.9775	0.9474	0.8254
44/100	0.0757	0.9771	1.0239	0.8212
45/100	0.0731	0.9792	0.9554	0.8292
46/100	0.0689	0.9804	1.0073	0.8230
47/100	0.0682	0.9809	1.0359	0.8228
48/100	0.0651	0.9827	1.061	0.8242
49/100	0.0615	0.9843	0.9835	0.8276
50 / 100	0.0598	0.9836	1.0075	0.8288
51 / 100	0.0583	0.9859	0.977	0.8296
52 / 100	0.0542	0.9857	1.2336	0.8098
53 / 100	0.0519	0.9882	0.9897	0.8340
54 / 100	0.0492	0.9881	1.223	0.8200

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
55 / 100	0.0468	0.9891	1.0489	0.8264
56 / 100	0.0456	0.9905	1.2621	0.8132
57 / 100	0.0434	0.9901	1.1457	0.8260
58 / 100	0.0396	0.9907	1.0934	0.8302
59 / 100	0.0395	0.9911	1.1126	0.8340
60 / 100	0.0365	0.9918	1.3168	0.8164
61 / 100	0.0354	0.9921	1.2735	0.8246
62 / 100	0.032	0.9936	1.432	0.8196
63 / 100	0.0308	0.9937	1.2756	0.8330
64 / 100	0.0297	0.9941	1.179	0.8366
65 / 100	0.0272	0.9946	1.3122	0.8308
66 / 100	0.0253	0.9952	1.4634	0.8196
67 / 100	0.0247	0.9955	1.4153	0.8226
68 / 100	0.0233	0.9961	1.2986	0.8330
69 / 100	0.0211	0.996	1.3651	0.8308
70 / 100	0.0215	0.9967	1.4366	0.8276
71 / 100	0.0185	0.9968	1.3978	0.8314
72 / 100	0.0170	0.9973	1.6028	0.8226
73 / 100	0.0163	0.9971	1.3698	0.8302
74 / 100	0.0154	0.9976	1.8460	0.8126
75 / 100	0.0142	0.9975	1.5969	0.8268
76 / 100	0.0135	0.9979	1.6746	0.8228
77 / 100	0.0120	0.9981	1.5466	0.8332
78 / 100	0.0114	0.9983	1.6801	0.8258
79 / 100	0.0105	0.9984	1.5240	0.8332
80 / 100	0.0094	0.9990	1.7059	0.8266
81 / 100	0.0088	0.9989	2.1167	0.8044
82 / 100	0.0076	0.9991	1.6928	0.8266
83 / 100	0.0073	0.9993	1.7823	0.8276
84 / 100	0.0060	0.9992	1.8105	0.8294
85 / 100	0.0061	0.9994	2.0048	0.8260
86 / 100	0.0058	0.9993	2.0594	0.8232
87 / 100	0.0047	0.9994	2.0616	0.8224
88 / 100	0.0041	0.9995	2.0568	0.8236
89 / 100	0.0038	0.9997	2.1799	0.8214
90 / 100	0.0034	0.9997	2.2058	0.8304
91 / 100	0.0030	0.9999	2.3742	0.8430
92 / 100	0.0027	0.9999	1.8925	0.8674
93 / 100	0.0025	0.9996	2.2388	0.8762
94 / 100	0.0024	0.9997	2.4344	0.8886
95 / 100	0.0020	0.9998	2.4973	0.8986
96 / 100	0.0019	0.9999	1.8693	0.9014
97 / 100	0.0014	0.9999	2.3205	0.9284
98 / 100	0.0011	1	2.3119	0.9394
99 / 100	9.97E-04	0.9999	2.6476	0.9510
100 / 100	8.95E-04	1	2.4416	0.9755

The training and validation results of the VGG-19C model are shown in Table 4 and Figure 3.

Table 4. Training and validation results of the VGG-19C model

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
1/100	0.3349	0.8756	0.5688	0.7760
2/100	0.2538	0.9036	0.5303	0.7902
3/100	0.2343	0.9118	0.5928	0.7926

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
4/100	0.2216	0.9153	0.5179	0.8126
5/100	0.2112	0.9194	0.5443	0.8152
6/100	0.2041	0.9201	0.5973	0.8078
7/100	0.1989	0.9216	0.6048	0.8114
8/100	0.1954	0.9244	0.5916	0.8106
9/100	0.1934	0.9232	0.6603	0.8072
10/100	0.1882	0.9272	0.6880	0.8078
11/100	0.1861	0.9276	0.7364	0.8040
12/100	0.1840	0.9285	0.7216	0.8048
13/100	0.1820	0.9293	0.7575	0.8028
14/100	0.1787	0.9296	0.7229	0.8082
15/100	0.1755	0.9314	0.6675	0.8088
16/100	0.1723	0.9315	0.7160	0.8098
17/100	0.1713	0.9352	0.6333	0.8176
18/100	0.1685	0.9366	0.7949	0.8054
19/100	0.1634	0.9395	0.6953	0.8166
20/100	0.164	0.9361	0.7947	0.8088
21/100	0.1606	0.9391	0.7571	0.8132
22/100	0.1567	0.9400	0.8629	0.8062
23/100	0.1534	0.9420	0.8066	0.8112
24/100	0.1525	0.9412	0.8630	0.8084
25/100	0.1501	0.9441	0.9439	0.8062
26/100	0.1476	0.9451	0.8715	0.8088
27/100	0.1458	0.9442	0.9569	0.8060
28/100	0.1431	0.9485	0.9511	0.8046
29/100	0.1405	0.9480	0.8984	0.8132
30/100	0.1340	0.9499	0.9540	0.8108
31/100	0.1356	0.9501	0.7550	0.8214
32/100	0.1306	0.9545	0.8648	0.8160
32/100	0.1306	0.9545	0.8648	0.8160
33/100	0.1298	0.9534	0.8689	0.8154
34/100	0.1268	0.9569	0.8163	0.8228
35/100	0.1240	0.9557	0.8959	0.8158
36/100	0.1236	0.9577	1.0246	0.8094
37/100	0.1190	0.9587	0.9895	0.8158
38/100	0.1172	0.9602	1.1166	0.8060
39/100	0.1120	0.9610	1.0929	0.8104
40/100	0.1097	0.9627	0.9209	0.8230
41/100	0.1079	0.9659	1.1096	0.8064
42/100	0.1034	0.9657	1.0814	0.8146
43/100	0.0999	0.9679	1.1010	0.8168
44/100	0.0974	0.9691	1.0124	0.8216
45/100	0.0970	0.9700	1.1169	0.8188
46/100	0.0944	0.9693	1.1937	0.8122
47/100	0.0888	0.9709	1.0374	0.8246
48/100	0.089	0.9709	1.0962	0.8198
49/100	0.0861	0.9733	1.0966	0.8228
50/100	0.0841	0.9741	0.9950	0.8276
51/100	0.0791	0.9769	1.1169	0.8194
52/100	0.0781	0.9764	1.1598	0.8220
53/100	0.0740	0.9795	1.2084	0.8198
54/100	0.0732	0.9800	1.202	0.8206
55/100	0.0696	0.9814	1.3061	0.8146
56/100	0.0659	0.9821	1.306	0.8176
57/100	0.0653	0.9816	1.3766	0.8148

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
58/100	0.0636	0.9825	1.4566	0.8102
59/100	0.0592	0.9851	1.3576	0.8182
60/100	0.0573	0.9856	1.3334	0.8216
61/100	0.0557	0.9868	1.4545	0.8126
62/100	0.0523	0.9859	1.3008	0.8260
63/100	0.0523	0.9883	1.4315	0.8186
64/100	0.0481	0.9876	1.2682	0.8300
65/100	0.0471	0.9884	1.4398	0.8244
66/100	0.0427	0.9901	1.5312	0.8148
67/100	0.0392	0.9908	1.5541	0.8224
68/100	0.0396	0.9903	1.5583	0.8220
69/100	0.0381	0.9915	1.5567	0.8200
70/100	0.0351	0.9927	1.693	0.8142
71/100	0.0347	0.9921	1.6603	0.8176
72/100	0.0331	0.9934	1.4412	0.8264
73/100	0.0313	0.9934	1.6319	0.8232
74/100	0.0280	0.9948	1.5561	0.8244
75/100	0.0258	0.9948	1.6204	0.8244
76/100	0.0250	0.9951	1.7106	0.8198
77/100	0.0255	0.9948	1.7369	0.8248
78/100	0.0226	0.9949	1.707	0.8250
79/100	0.0204	0.9966	1.7334	0.8262
80/100	0.0183	0.9969	1.9623	0.8112
81/100	0.0188	0.9973	1.7961	0.8242
82/100	0.0169	0.9967	1.7671	0.8254
83/100	0.0172	0.9970	1.7826	0.8264
84/100	0.0151	0.9973	1.9336	0.8216
85/100	0.0126	0.9982	2.1171	0.8186
86/100	0.0128	0.9980	1.9428	0.8228
87/100	0.0120	0.9984	2.2851	0.8266
88/100	0.0102	0.9988	1.9325	0.8346
89/100	0.0105	0.9981	1.901	0.8468
90/100	0.0097	0.9991	2.0587	0.8564
91/100	0.0088	0.9990	2.1058	0.8642
92/100	0.0089	0.9987	2.3508	0.8760
93/100	0.0075	0.9988	1.9600	0.8812
94/100	0.0070	0.9993	2.0539	0.8960
95/100	0.0067	0.9992	2.2851	0.9116
96/100	0.0058	0.9996	2.2477	0.9250
97/100	0.0051	0.9993	2.3583	0.9308
98/100	0.0057	0.9991	2.5974	0.9462
99/100	0.0048	0.9997	2.4989	0.9752
100/100	0.0043	0.9996	2.1695	0.9883

The training and validation results of the VGG-19D model are shown in Table 5 and Figure 4.

Table 5. Training and validation results of the VGG-19D model

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
1/100	0.3108	0.8888	0.5443	0.7724
2/100	0.2398	0.9063	0.4871	0.8112
3/100	0.2203	0.9149	0.5840	0.7914
4/100	0.2084	0.9182	0.5894	0.7964
5/100	0.2002	0.9203	0.5687	0.8040
6/100	0.1953	0.9249	0.5676	0.8074

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
7/100	0.1885	0.9250	0.5618	0.8062
8/100	0.1839	0.9257	0.5976	0.8058
9/100	0.1781	0.9295	0.6025	0.8054
10/100	0.1766	0.9300	0.6301	0.8068
11/100	0.1726	0.9312	0.5985	0.8092
12/100	0.1692	0.9344	0.6517	0.8068
13/100	0.1655	0.9364	0.6528	0.8068
14/100	0.1606	0.9389	0.6385	0.8102
15/100	0.1588	0.9386	0.6722	0.8078
16/100	0.1538	0.9398	0.6290	0.8128
17/100	0.1505	0.9449	0.6300	0.8150
18/100	0.1481	0.9435	0.6189	0.8150
19/100	0.1457	0.9453	0.7476	0.8086
20/100	0.1419	0.9468	0.6346	0.8222
21/100	0.1403	0.9484	0.7247	0.8140
22/100	0.1334	0.9491	0.7873	0.8100
23/100	0.1313	0.9524	0.7193	0.8142
24/100	0.1270	0.9545	0.7466	0.8156
25/100	0.1261	0.9531	0.7458	0.8184
26/100	0.1214	0.9561	0.8872	0.8058
27/100	0.1185	0.959	0.6927	0.8198
28/100	0.1162	0.9588	0.9095	0.8076
29/100	0.1141	0.9601	0.6847	0.8260
30/100	0.1090	0.9606	0.8041	0.8160
31/100	0.1072	0.9606	1.0424	0.8050
32/100	0.1035	0.9656	0.9668	0.8104
33/100	0.1002	0.9652	0.8691	0.8146
64/100	0.0223	0.9959	1.3825	0.8258
65/100	0.0203	0.996	1.601	0.8152
66/100	0.0188	0.997	1.4347	0.8230
67/100	0.0176	0.9975	1.4804	0.8256
68/100	0.0160	0.9978	1.4836	0.8252
69/100	0.0152	0.9976	1.5674	0.8208
70/100	0.0140	0.9981	1.5481	0.8232
71/100	0.0130	0.9984	1.5227	0.8214
72/100	0.0121	0.9983	1.417	0.8314
73/100	0.0111	0.9989	1.4488	0.8360
74/100	0.0101	0.9985	1.4712	0.8300
75/100	0.0093	0.9988	1.5764	0.8264
76/100	0.0084	0.9989	1.6357	0.8268
77/100	0.0074	0.9989	1.8639	0.8194
78/100	0.0077	0.9991	1.7494	0.8242
79/100	0.0064	0.9993	1.9100	0.8218
80/100	0.0058	0.9993	1.9576	0.8158
81/100	0.0056	0.9993	2.0104	0.8204
82/100	0.0042	0.9996	1.9334	0.8230
83/100	0.0043	0.9996	1.9391	0.8234
84/100	0.0040	0.999	2.0536	0.8198
85/100	0.0031	0.9997	1.9390	0.8280
86/100	0.0024	0.9997	1.8474	0.8332
87/100	0.0023	0.9998	1.9956	0.8246
88/100	0.0020	0.9999	2.0974	0.8232
89/100	0.0020	0.9997	2.1888	0.8224
90/100	0.0013	0.9999	2.0936	0.8272
91/100	0.0014	0.9999	2.1994	0.8358

Epoch	Loss	Accuracy	Validation loss	Validation accuracy
92/100	0.0014	0.9998	1.9263	0.8440
93/100	0.0010	1	2.3412	0.8758
94/100	0.0009	0.9999	2.3328	0.8952
95/100	0.0007	0.9999	2.5126	0.9046
96/100	0.0006	1	2.5790	0.9248

The summary of both training and validation results and test results of the four models is shown in Table 6 and Table 7, respectively.

Table 6. Summary of training and validation results

Models	Loss	Accuracy	Validation loss	Validation accuracy
VGG-19A	2.14E-05	100	2.98	97.02
VGG-19B	8.95E-04	100	2.45	97.55
VGG-19C	0.0043	99.96	1.17	98.83
VGG-19D	0.00033169	100	2.47	97.53

Table 7. Summary of test results

Networks	Accuracy	Loss
VGG-19A	97.65	2.35
VGG-19B	98.13	1.87
VGG-19C	99.16	0.84
VGG-19D	98.95	1.05

Performance of VGG-19A, VGG-19B, VGG-19C, and VGG-19D models was evaluated using several standard performance evaluation metrics, such as top-1 accuracy, threshold operation, quality test, fake face test, and overall test. The four VGG-19 algorithms were trained and tested on the six datasets. The datasets had 2,337 real subjects, and their fake faces were generated using wrapped photo, cut-photo, mask and video attacks, respectively. The low, normal, and high quality imaging was considered. The datasets provided 80,000 face images, with 40,000 real faces and 40,000 fake ones.

The top-1 accuracy measured the proportion of face images with the predicted label matching the single target label. The output class with the highest probability was considered. The results of class output probability as well as real and fake face output are depicted in Table 8 and Table 9.

Table 8. Probability of class output

	Positive	Negative
True	1	1
False	0	0

Table 9. Real and fake face output

Input	Output	
Real face	Real face (true positive)	Fake face (false negative)
Fake face	Fake face (true negative)	Real face (false positive)

The top-1 accuracy result of VGG-19A, VGG-19B, VGG-19C, and VGG-19D models is depicted in Table 10.

It can be seen from Table 10 that the VGG-19C obtained an average top-1 accuracy of 99%. Dropout rate of 0.5 was utilized for additional regularization after every max-pooling layer in VGG-19C, which reduced the amount of overfitting in the model, thus causing a strong generalization ability.

Threshold operation accuracy considered only the probability of real face class. The probability of the true class obtained in the output of the VGG-19 models was compared with a set of threshold values to determine whether the output probability was higher than the corresponding threshold. It can be observed from Table 11 that VGG-19C

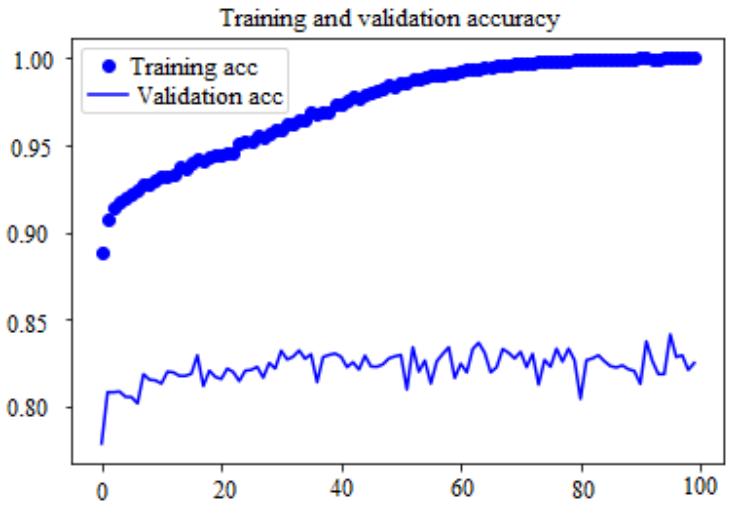

Figure 2. Training and validation accuracy of the VGG-19B model

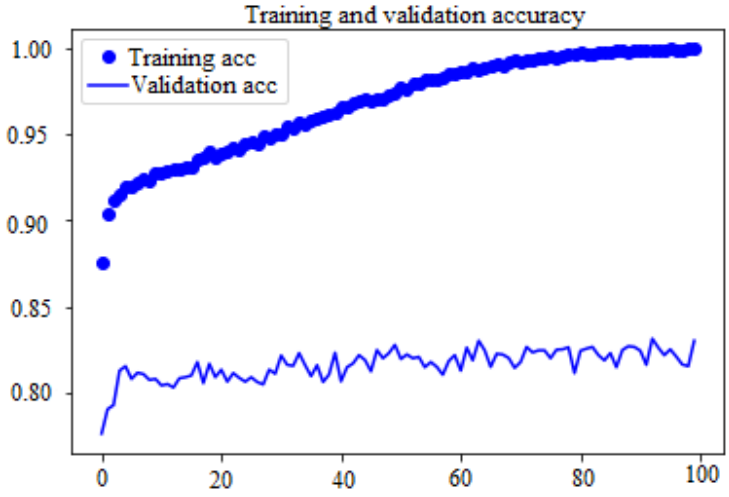

Figure 3. Training and validation accuracy of the VGG-19C model

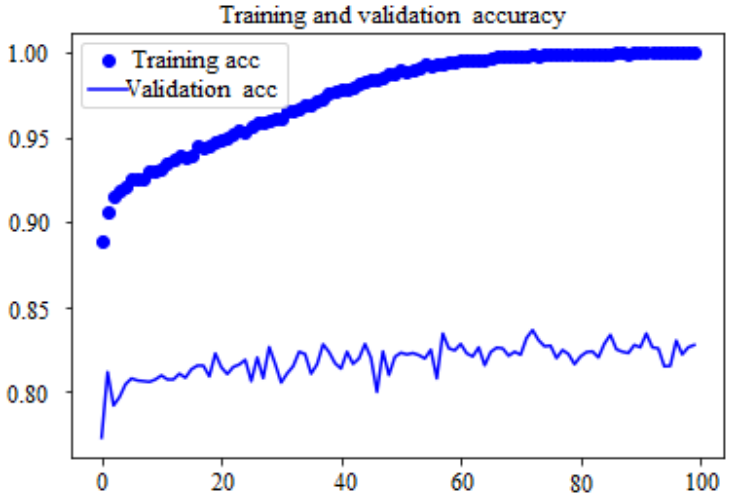

Figure 4. Training and validation accuracy of the VGG-19D model

Table 10. Top-1 accuracy

SAD models	Top-l accuracy
VGG-19A	97%
VGG-19B	98%
VGG-19C	99%
VGG-19D	98%

Table 11. Threshold operation accuracy

SAD models	Threshold operation accuracy	EER
VGG-19A	85%	7%
VGG-19B	82%	7%
VGG-19C	99%	0.4 %
VGG-19D	90%	8%
CNN + SVM [30]	81%	7.49%
VGG-11 [31]	89%	5%
VGG-16 [32]	97%	0.67%

has achieved the best performance. Compared with the state-of-the-art approaches [30–32], VGG-19C achieved an average test accuracy of 99% with the lowest EER of 0.4%.

This quality test was used to evaluate the performance of the SAD model given that the input image quality was fixed. The images of each subject were captured concurrently using three cameras with low, normal, and high quality. The SVD face detector was used to classify the image quality, and the quality test result on the datasets is depicted in Table 12.

Table 12. Quality test accuracy

SAD models	Normal	Low	High
VGG-19A	91%	84%	78%
VGG-19B	94%	86%	80%
VGG-19C	97%	94%	84%
VGG-19D	95%	87%	82%

The fake face test was used to evaluate the performance of the face SAD model given that the fake face image types were fixed. The fake face images were in wrapped photo, cut photo, video, and mask attacks. The significance of this metric was to have as many as possible fake face image samples to train and test the VGG-19 model. The result is depicted in Table 13.

Table 13. Fake face test accuracy

SAD models	Wrapped photo attack	Cut photo attack	Video attack	Mask attack
VGG-19A	79%	80%	82%	89%
VGG-19B	89%	82%	82%	87%
VGG-19C	98%	87%	89%	88%
VGG-19D	78%	90%	90%	90%

It can be observed from Table 12 that the VGG-19C model has obtained an overall high classification rate, compared with VGG-19A, VGG-19B, and VGG-19D. Similarly, VGG-19C has obtained an overall high accuracy for all four types of attacks in fake face test, as shown in Table 13.

The overall test was used to evaluate the general performance of the face SAD model by combining all the data. Table 14 compares the quality test using the proposed VGG-19 model for the face SAD model with other state-of-the-art approaches using CNN with SVM, VGG-11, and VGG-16 for developing the model.

4 Discussion

The VGG-19 base architecture was designed for different image classification types. In order to achieve the best performance in classifying human face images, VGG-19B, VGG-19C, and VGG-19D were derived from the

Table 14. Overall test accuracy

SAD models	Low	Normal	High
VGG-11 [31]	94%	94%	82%
CNN + SVM [30]	75%	83%	90%
VGG-16 [32]	91%	90%	89%
VGG-19C	97%	96%	95%

base architecture and trained on the same normalized extracted features. The multi-ethnicity face datasets across the entire globe were used for the training of the SAD model.

Extraction of RGB and deep network features from the face images produced more distinct traits for the training and validation of the face SAD model. The normalized extracted features were used to train, validate, and test VGG-19A, VGG-19B, VGG-19C, and VGG-19D models. VGG-19A, VGG-19B, and VGG-19D obtained 100% accuracy while VGG-19C achieved 99.96% accuracy. However, VGG-19C was the best SAD model because it had the best validation accuracy of 98.83%. VGG-19C achieved 0.4% EER and an improvement of 6% in overall test compared with the state-of-the-art approaches, after being evaluated using top-1 accuracy, threshold operation accuracy, quality test, fake face test, overall test and benchmark to CNN with SVM, VGG-11, and VGG-16.

5 Conclusions

Face datasets were collected from CASIA, NUAA, OULU, WMCA, 3DMAD, and CASIA-Face-Africa, and pre-processed using SVD. Dropout regularization technique was added to neural network layers to overcome overfitting. VGG-19 architecture was used to extract features from RGB and deep network to classify real and fake face images. The designed neural networks were implemented with TensorFlow 2.0 framework. The extracted features were normalized, and used to train the four VGG-19 neural networks. The trained networks were tested with unknown face datasets, which showed that VGG-19C was the best face SAD model with 99.96% training accuracy, 98.83% validation accuracy, and 99.16% testing accuracy. In addition, VGG-19C was evaluated using top-1 accuracy, threshold operation accuracy, quality test, fake face test, overall test and benchmark to face SAD models of CNN with SVM, VGG-11, and VGG-16, which showed that VGG-19C achieved 0.4% EER, and an improvement of 6% in overall test compared with the state-of-the-art approaches.

Although the study of face SAD has been progressing tremendously using new algorithms and techniques, some open research issues need to be addressed to develop a robust face SAD model for real-time face verification and authentication applications. The most common issue with all face SAD systems is their generalization ability in unconstrained scenes. Apart from face biometric verification and authentication for access control, other potential applications of face SAD need to be explored, such as recognizing live people and photos for self-driving assistance and robot navigation.

Data Availability

The study used CASIA, NUAA, WMCA, OULU, 3DMAD, and CASIA-Face-Africa datasets. The datasets were collected through a research agreement form after which the datasets were released. This procedure was followed to collect the face datasets from owners of the spoofing attack detection datasets:

1. An email was sent for use of the spoofing attacks datasets for academic research purpose using my Babcock student email address.
2. The license agreement form was filled in conjunction with my Thesis supervisors.
3. The filled license agreement form was scanned and sent back.
4. The face datasets were downloaded to my google drive for training, validating and testing of the face SAD models.
5. Extraction of face datasets was done using ZIP Extractor which is compatible with Google Drive.

Acknowledgements

My profound and honest appreciation goes to my research supervisors, Professor O. Awodele; Dr. M. Agbaje; and Dr. A. Ajayi of the School of Computing and Engineering Sciences, Babcock University, Ilishan, for availing me a most conducive environment to carry out this research and for the priceless supervision they provided in the course of this program. I am deeply inspired, motivated, touched, and cared for by their patience, professional comments, dedication, and input. They played key roles that culminated in the accomplishment of this thesis.

I also acknowledge the immense backing of the entire staff of the Department of Computer Science. I thank Professor S. Idowu, Dean School of Computing and Engineering Sciences, Dr. S. Kuroyo, Head of Department Computer Science, Professor S. Okolie, PG School Coordinator, Dr E.E. Onuiri, PG Department Coordinator, Dr S.

Maitanmi, School of Sciences Methodologist, Professors A. Ogonna; M. Eze; A. Adebayo, and Drs. O. Ebiesuwa; A. Omotunde, T. Adigun, U. Nzenwatta, C. Ajaegbu, F. Ayankoya, O. Akande, and A. Izang. I thank them for making the Department a warm and conducive environment for learning.

I am particularly indebted to my parents: Mr. Ayanwola Benjamin (late) and Mrs. Mary Ayanwola (nee Oladele) for their love, prayers, care, and sacrifice to ensure that I have access to quality education and a secured future. I am very much thankful to my beautiful wife, Mrs. Dolapo Bosede Ayanwola (nee Agboola), and my lovely God-given children: Ayanwola Faith Oluwanifemi, Ayanwola David Oluwatimilehin, Ayanwola Emmanuel Jesutofunmi, and Ayanwola Joseph Eri-Oluwa for their love, understanding, prayers, and continuing support throughout the program. My special thanks go to my pastor and father in the Lord, Pastor Noruwa Edokpolo, the Provincial Pastor of Lagos Province 77, Redeemed Christian Church of God, for all the opportunities given to me to progress academically.

I am grateful for the administrative assistance, information shared, printing, and editorial support received from some members of the Babcock University community. I particularly acknowledge the support of Mr. Seun Idowu, Mr. Martins Nkume, and Mrs. Jane Chukwuemeka.

Finally, my deepest thanks go to my siblings, entire family, and friends who supported me in one way or the other through the course of this work.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] B. Choudhury, M. Haldar, B. Issac, V. Raman, and P. Then, "A survey on biometrics and cancelable biometrics systems," *Int. J. Image Graph.*, vol. 18, no. 1, pp. 1–39, 2018. http://dx.doi.org/10.1142/S0219467818500067

[2] D. Kalenichenko, J. Philbin, and F. Schroff, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 07-12, 2015, pp. 815–823. https://doi.org/10.1109/CVPR.2015.7298682

[3] C. Busch and R. Ramachandra, "Presentation attack detection methods for face recognition systems: A comprehensive survey," *ACM Comput. Surv.*, vol. 50, no. 1, pp. 1–37, 2017. http://dx.doi.org/10.1145/3038924

[4] R. H. Deng, Y. Li, K. Xu, Y. J. Li, and Q. Yan, "Understanding osn-based facial disclosure against face authentication systems," in *Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security*, Kyoto, Japan, June 4-6, 2014, pp. 413–424. https://doi.org/10.1145/2590296.2590315

[5] X. Y. Tan, Y. Li, J. Liu, and L. Jiang, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," in *Computer Vision – ECCV 2010. ECCV 2010. Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer, 2010, pp. 504–517. https://doi.org/10.1007/978-3-642-15567-3_37

[6] A. Orjiude, "Eight banks lose n1.9bn to fraud in one year," Punch Newspaper, 2021. https://punchng.com/eight-banks-lose-n1-9bn-to-fraud-in-one-year/

[7] Z. F. Li, Y. Qiao, K. P. Zhang, and Z. P. Zhang, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, 2016. https://doi.org/10.1109/LSP.2016.2603342

[8] Z. W. Zhang, J. J. Yan, S. F. Liu, Z. Li, Z. Lei, D. Yi, and S. Z. Li, "A face anti-spoofing database with diverse attacks," in *5th IAPR International Conference on Biometrics (ICB)*, New Delhi, India, March 29, 2012 - April 01, 2012, pp. 26–31. https://doi.org/10.1109/ICB.2012.6199754

[9] T. Mäenpää, T. Ojala, and M. Pietikäinen, "Multiresolution grayscale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, 2002. https://doi.org/10.1109/TPAMI.2002.1017623

[10] J. Komulainen, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using texture and local shape analysis," *IET Biometrics*, vol. 1, no. 1, pp. 3–10, 2012. http://dx.doi.org/10.1049/iet-bmt.2011.0009

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of IEEE Computing Society Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA, 2005, pp. 886–893. https://doi.org/10.1109/CVPR.2005.177

[12] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, 1996. https://doi.org/10.1109/34.531803

[13] D. Gragnaniello, G. Poggi, C. Sansone, and L. Verdoliva, "An investigation of local descriptors for biometric spoofing detection," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 4, pp. 849–863, 2015. https://doi.org/10.1109/TIFS.2015.2404294

[14] H. X. Li, Z. Lin, X. H. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 07-12, 2015, pp. 5325–5334. https://doi.org/10.1109/CVPR.2015.7299170

[15] L. Li, X. Y. Feng, X. Y. Jiang, Z. Q. Xia, and A. Hadid, "Face anti-spoofing via deep local binary patterns," in *Proceedings of IEEE International Conference on Image Processing*, Beijing, China, September 17-20, 2017, pp. 101–105. https://doi.org/10.1109/ICIP.2017.8296251

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint*, 2014. https://doi.org/10.48550/arXiv.1409.1556

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F. F. Li, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015. https://doi.org/10.1007/s11263-015-0816-y

[18] Z. Zhang, C. Jiang, X. Zhong, C. Song, and Y. Zhang, "Two-stream convolutional networks for multi-frame face anti-spoofing," *arXiv preprint*, 2021. https://doi.org/10.48550/arXiv.2108.04032

[19] N. A. Taha, T. Hassan, and M. A. Younus, "Face spoofing detection using deep CNN," *Turk. J. Comput. Math. Educ.*, vol. 12, no. 13, pp. 4363–4373, 2021.

[20] S. Shekhar, A. Patel, M. Haloi, and A. Salim, "An ensemble model for face liveness detection," *arXiv preprint*, 2022. https://doi.org/10.48550/arXiv.2201.08901

[21] Z. T. Yu, C. X. Zhao, K. H. M. Cheng, X. M. Cheng, and G. Y. Zhao, "Flexible-modal face anti-spoofing: A benchmark," *arXiv preprint*, 2022. https://doi.org/10.48550/arXiv.2202.08192

[22] Y. Bian, P. Zhang, J. J. Wang, C. M. Wang, and S. L. Pu, "Learning multiple explainable and generalizable cues for face anti-spoofing," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Singapore, Singapore, May 23-27, 2022, pp. 2310–2314. https://doi.org/10.1109/icassp43922.2022.9747677

[23] Y. K. Sharma, S. P. Patil, and R. D. Patil, "Deep transfer learning for face spoofing detection," *IOSR J. Comput. Eng.*, vol. 22, no. 5, pp. 16–20, 2020. http://dx.doi.org/10.9790/0661-2205031620

[24] J. Z. Guo, X. Y. Zhu, J. C. Xiao, Z. Lei, G. X. Wan, and S. Z. Li, "Improving face anti-spoofing by 3d virtual synthesis," in *2019 International Conference on Biometrics (ICB)*, Crete, Greece, February 10, 2019, pp. 1–8. https://doi.org/10.1109/ICB45273.2019.8987415

[25] Z. H. Ming, Z. T. Yu, M. Al-Ghadi, M. Visani, M. M. Luqman, and J. C. Burie, "Vitranspad: Video transformer using convolution and self-attention for face presentation attack detection," in *2022 IEEE International Conference on Image Processing (ICIP)*, Bordeaux, France, 2022, pp. 4248–4252. https://doi.org/10.1109/ICIP46576.2022.9897560

[26] Y. H. Zhang, Y. C. Wu, Z. F. Yin, J. Shao, and Z. W. Liu, "Robust face anti-spoofing with dual probabilistic modelling," *arXiv preprint*, 2022. https://doi.org/10.48550/arXiv.2204.12685

[27] S. L. Yang, W. Wang, C. Y. Xu, B. Peng, and J. Dong, "Exposing fine-grained adversarial vulnerability of face anti-spoofing models," *arXiv preprint*, 2023. http://dx.doi.org/10.48550/arXiv.2205.14851

[28] N. Sergievskiy, R. Vlasov, and R. Trusov, "Generalizable method for face anti-spoofing with semi-supervised learning," *arXiv preprint*, 2022. http://dx.doi.org/10.48550/arXiv.2206.06510

[29] S. Chen, T. Xu, Z. Feng, X. Wu, and J. Kittler, "Face anti-spoofing with local difference network and binary facial mask supervision," *J. Electron. Imaging*, vol. 31, no. 1, pp. 1–18, 2022. http://dx.doi.org/10.1117/1.JEI.31.1.013007

[30] J. Yang, Z. Lei, and S. Z. Li, "Learn convolutional neural network for face anti-spoofing," *arXiv preprint*, 2014. https://doi.org/10.48550/arXiv.1408.5601

[31] M. P. Lai, M. M. Li, and A. R. U. Yasar, "Deep learning for face anti-spoofing: An end-to-end approach," in *2017 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, 2017, pp. 195–200. https://doi.org/10.23919/SPA.2017.8166863

[32] D. A. Muhtasim, M. I. Pavel, and S. Y. Tan, "A patch-based CNN built on the vgg-16 architecture for real-time facial liveness detection," *Sustainability*, vol. 14, no. 16, pp. 1–11, 2022. http://dx.doi.org/10.3390/su141610024