Hindawi
Security and Communication Networks
Volume 2021, Article ID 9939232, 11 pages
https://doi.org/10.1155/2021/9939232

Research Article

Face Antispoofing Method Using Color Texture Segmentation on FPGA

Youngjun Moon[1], **Intae Ryoo**[1] **and Seokhoon Kim**[2,3]

[1]*Department of Computer Engineering, Kyung Hee University, Yongin-si, Gyeonggi-do 17104, Republic of Korea*
[2]*Department of Software Convergence, Soonchunhyang University, Asan-si, Chungcheongnam-do 31538, Republic of Korea*
[3]*Department of Computer Software Engineering, Soonchunhyang University, Asan-si, Chungcheongnam-do 31538, Republic of Korea*

Correspondence should be addressed to Intae Ryoo; itryoo@khu.ac.kr and Seokhoon Kim; seokhoon@sch.ac.kr

Received 4 March 2021; Revised 5 April 2021; Accepted 29 April 2021; Published 10 May 2021

Academic Editor: Beijing Chen

User authentication for accurate biometric systems is becoming necessary in modern real-world applications. Authentication systems based on biometric identifiers such as faces and fingerprints are being applied in a variety of fields in preference over existing password input methods. Face imaging is the most widely used biometric identifier because the registration and authentication process is noncontact and concise. However, it is comparatively easy to acquire face images using SNS, etc., and there is a problem of forgery via photos and videos. To solve this problem, much research on face spoofing detection has been conducted. In this paper, we propose a method for face spoofing detection based on convolution neural networks using the color and texture information of face images. The color-texture information combined with luminance and color difference channels is analyzed using a local binary pattern descriptor. Color-texture information is analyzed using the Cb, S, and V bands in the color spaces. The CASIA-FASD dataset was used to verify the proposed scheme. The proposed scheme showed better performance than state-of-the-art methods developed in previous studies. Considering the AI FPGA board, the performance of existing methods was evaluated and compared with the method proposed herein. Based on these results, it was confirmed that the proposed method can be effectively implemented in edge environments.

1. Introduction

Recently, authentication systems based on biometric information have been applied to various mobile devices such as smartphones, and many users perform identity authentication using facial or fingerprint information instead of the existing password input methods. In addition, biometric authentication is being applied to bank transactions and mobile payment applications. As a result, researchers are greatly interested in developing high-performance authentication systems.

Among user biometric information, face images are the most widely used biometric identifier because the associated registration and authentication processes are noncontact and concise. However, face images are very easy to acquire using social networks, etc., and are vulnerable against various spoofing techniques, including printed photos and video replay. To solve this problem, research utilizing software solutions have become popular, rather than antispoofing hardware solutions using additional sensors. These software approaches can be classified into motion-based methods and texture-based methods [1].

The motion-based counterfeit face detection method measures eye/head movement, eye blinking, and changes in facial expression [2, 3]. In the case of counterfeit face detection methods utilizing eyes, note that a still face such as in a photograph does not exhibit eye blinking or pupil movement, as opposed to real human faces which exhibit

relatively large amounts of movement over time. This method is very simple and fast. However, this method classifies a spoofing face using only eye movement and thus cannot defend against simple attack variations that focus on and accurately emulate the eye area based on a photo.

The texture-based spoofing face detection method mainly uses lighting characteristics that appear differently between 2D plane and 3D stereoscopic objects or uses a fine texture difference between the spoofing face data and live face data through an external medium such as printing [4–8]. This method mainly uses a local image descriptor such as an LBP (local binary pattern) [9] to express differences in the texture characteristics between live and spoofing face images. Such texture-based methods have been actively researched due to the advantages of easy implementation and short detection times; however, these methods have difficulty classifying liveness faces in nonuniform images or images with large amounts of noise. Recently, researchers have been working on the detection of spoofing faces using convolutional neural networks (CNNs) [10, 11]. Since this method can effectively derive features through learning, its performance is improved over existing texture-based detection methods.

Although the field of spoofing face detection has developed tremendously, the existing methods mainly focus on the brightness information of face images. More specifically, other color information, which is similar to brightness information, is often overlooked in spoofing face detection. Therefore, by considering both color and brightness information of face images, a method was proposed that independently extracts texture features from the brightness space and color space of the face image using an LBP [12].

The difference between a real face and spoofing face is discriminated using a descriptor (such as an LBP) that encodes comparison results with respect to surrounding pixel values in a binary pattern at all pixel locations. However, since it is possible to produce high-resolution images, it is very difficult to distinguish detailed surface differences between real faces and spoofing faces using only pixel brightness.

In this paper, we propose a liveness face detection method based on a convolutional neural network utilizing the color and texture information of a face image. The proposed method analyzes the combined color-texture information in terms of its luminance and color difference channels using an LBP descriptor. For color-texture information analysis, the Cb, S, and H bands are used from the color spaces.

The rest of the paper is organized as follows. In Section 2, the related key technologies are illustrated. The proposed scheme for our color-texture-based antispoofing is presented in Section 3. Section 4 thoroughly presents the results and discussion. Finally, conclusions are presented in Section 5.

2. Related Works

2.1. Face Antispoofing. Conventional face antispoofing methods generally create spoofing patterns by extracting features from face images. Classic local descriptors such as LBP [13], SIFT [14], SURF [15], HOG [16], and DoG [17] are used to extract frame level functions, while methods such as dynamic texture [18], micromotion [19], and eye blinking [20] extract video features.

Recently, several deep learning-based methods have been studied to prevent face spoofing at the frame and video levels. In frame level methods [21–24], the pretrained CNN model is fine-tuned to extract features from the binary classification setup [25–27].

2.2. Color Spaces. RGB is a color space commonly used for sensing and displaying color images. However, its use in image analysis is typically limited because the three colors (red, green, and blue) are not separated according to luminance and color difference information. Thus, it is common to additionally convert the RGB information into YCbCr and HSV information before use. These two latter color spaces are based on luminance and chrominance information [28–31]. In particular, the YCbCr Color space separates RGB into luminance (Y), chrominance blue, and chrominance red. Similarly, the HSV color space uses the hue and saturation dimensions to define the color differences of the image, and the value dimension corresponds to the luminance.

2.3. LBP (Local Binary Pattern). LBPs [32, 33] are a feature developed for classifying image textures. Since then, LBPs have been used for face recognition. LBPs are a simple operation used for image analysis and recognition and are robust to changes in discrimination and lighting. Equation (1) is an LBP equation:

$$\mathrm{LBP}(p,r) = \sum_{p=1}^{p-1} s\left(g_p - g_c\right)2^p, \tag{1}$$

$$s(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Here, g_p ranges over the pixel values excluding the center pixel and g_c is the center pixel in equation (1). In Figure 1, P is the number of adjacent pixels and R is the radius of the circle. Figure 2 shows an example result of LBP operation applied to a real photo [34].

3. Proposed Scheme for Color-Texture-Based Antispoofing

The RGB color space contains three color components, red, green, and blue; the YCbCr color space contains brightness and saturation information, and the HSV color space contains three components: hue, saturation, and brightness. Each color space contains different information and has its own characteristics. RGB contains rich spatial information that most closely resembles the colors seen by humans, while the YCbCr and HSV color spaces contain information that is more sensitive to brightness. The RGB color space can be converted into HSV and YCbCr, and the specific calculation is as follows:

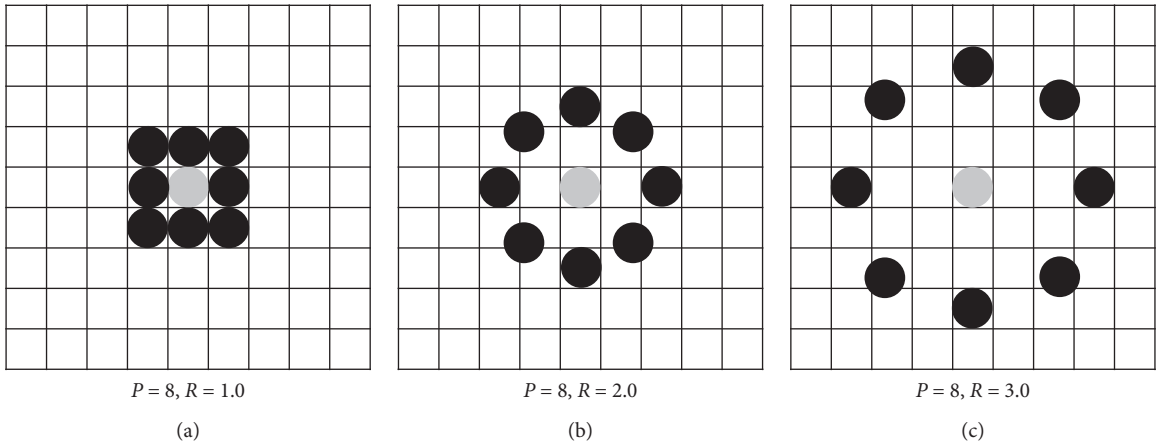

Figure 1: Example of a local binary pattern.

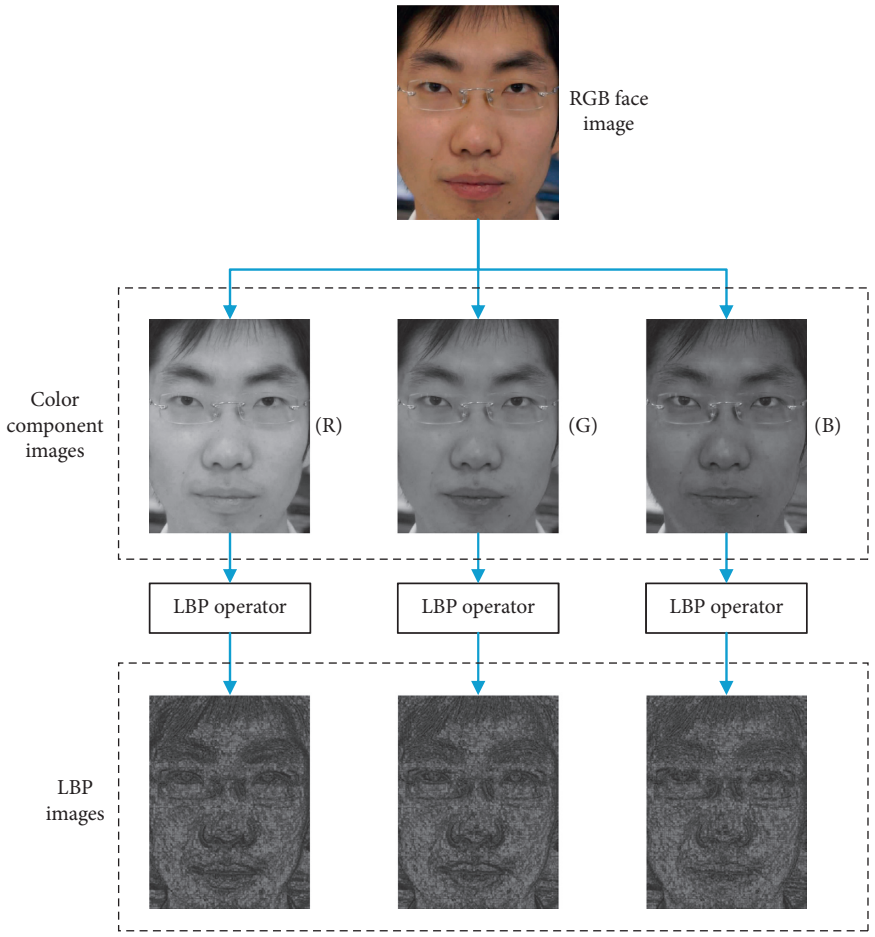

Figure 2: Visualization of LBP operation performed on each color band image.

$$V = \max(R, G, B),$$

$$S = \begin{cases} \dfrac{V - \min(R, G, B)}{V}, & \text{if } V \neq 0, \\ 0, & \text{if } V = 0, \end{cases}$$

$$H = \begin{cases} \dfrac{60(G - B)}{V - \min(R, G, B)}, & \text{if } V = R, \\ 120 + \dfrac{60(B - R)}{V - \min(R, G, B)}, & \text{if } V = G, \\ 240 + \dfrac{60(R - G)}{V - \min(R, G, B)}, & \text{if } V = B, \end{cases} \tag{3}$$

if $H < 0, H = H + 360$.

The YCbCr calculation formula is shown as

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B, \\ \text{Cb} &= 0.564(B - Y), \\ \text{Cr} &= 0.713(R - Y). \end{aligned} \tag{4}$$

In existing methods, RGB face images are converted into the YCbCr and HSV color spaces, and the spoofing images are classified by applying an LBP to each color space. However, this method increases the amount of computation because it uses a 6-channel color space. Figure 3 shows a conceptual diagram of the existing methods.

In this paper, we use a 3-channel color space consisting of Cb, S, and V, from which many facial features can be derived. The proposed method aims toward high-speed processing and robustness against lighting changes in face antispoofing. Figure 4 shows a conceptual diagram of the proposed scheme.

The advantages of this approach are summarized as follows:

(1) This proposed scheme reduces false detection by using a 3-channel color space in which sufficient facial feature information is expressed

(2) This proposed scheme uses less memory with fewer feature dimensions, thus enabling high-speed processing

4. Performance Evaluation

4.1. Train/Test Dataset. In this paper, we performed a spoofing face detection test using the CASIA Face Antispoofing Database (CASIA-FASD) [35] for performance evaluation. CASIA-FASD consists of real face videos and fake face videos acquired from 50 different users. The real face videos consist of three types of videos: low quality, medium quality, and high quality. Similarly, the fake face videos consist of three types of fake attack videos: printed photo attacks, cut photo attacks, and video relay attacks. Videos for 20 people are used for learning, while the remaining videos for 30 people are used for performance evaluation.

We extracted each frame from the CASIA-FASD dataset videos images for performance evaluation. In total, 4,577 live face images, 5,054 printed photo attack images, 2,368 cut photo attack images, and 4,429 video replay attack images were used for learning. In addition, 5,912 live face images, 7,450 printed photo attack images, 4,437 cut photo attack images, and 5,652 video replay attack images were used for evaluation. Table 1 shows detailed information on data partitioning of CASIA-FASD.

4.2. Experimental Setup. In this paper, we used FPGA for performance evaluation. We evaluated the performance of the proposed scheme by using the AI Accelerator of FPGA. The specifications of FPGA and the implemented board are shown in Figure 5.

Zynq® UltraScale+™ MPSoC devices provide 64-bit processor scalability while combining real-time control with soft and hard engines for graphics, video, waveform, and packet processing. Built on a common real-time processor and programmable logic-equipped platform, three distinct variants (dual application processor (CG) devices, quad application processor and GPU (EG) devices, and video codec (EV) devices) are included, creating numerous possibilities for various applications such as 5G wireless, next-generation ADAS, and industrial internet-of-things technologies [36].

Vitis AI is Xilinx's development stack for AI inference on Xilinx hardware platforms, including both edge devices and Alveo cards. It consists of an optimized IP, tools, libraries, models, and example designs. Vitis AI is designed with high efficiency and ease of use in mind, leading to great potential for AI acceleration on Xilinx FPGA and ACAP [37].

Face antispoofing detection uses AlexNet based on CNN. AlexNet is a basic model utilizing a convolutional layer, a pooling layer, and a fully connected layer [38].

AlexNet consists of five convolution layers and three full-connected (FC) layers, where the last FC layer uses softmax as an active function for category classification. Figure 6 shows Alexnet's CNN architecture.

4.3. Experimental Analysis Method. To evaluate the proposed scheme, we measured the HTER (Half Total Error Rate) in the CASIA-FASD dataset. The HTER is calculated using the false acceptance rate (FAR) and false rejection rate (FRR) in the attack dataset, both of which are defined below. The HTER calculation is given as follows [39]:

$$\text{HTER} = \frac{\text{FAR} + \text{FRR}}{2}. \tag{5}$$

The FAR [40] is a measure of how likely the biometric security system will incorrectly accept an access attempt by an unauthorized user. A system's FAR typically is defined as the ratio of the number of false acceptances divided by the number of identification attempts.

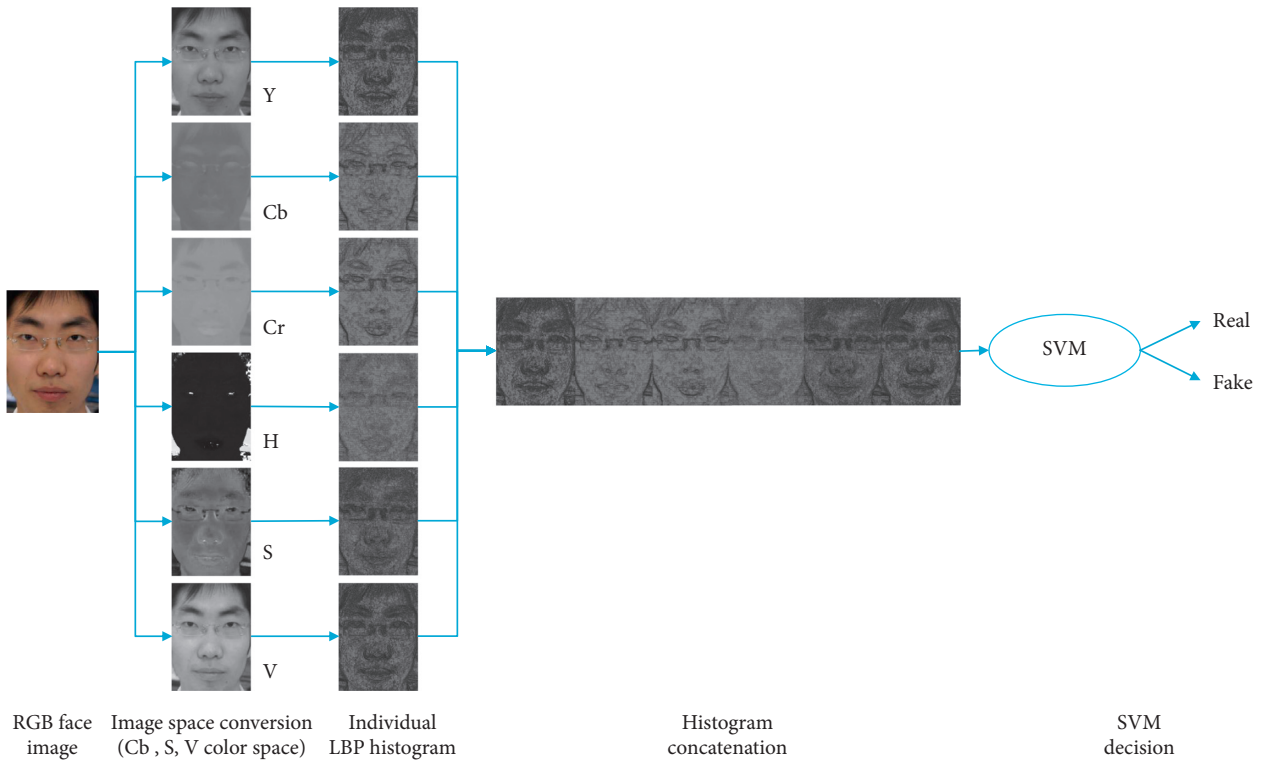

Figure 3: Conceptual diagram of existing methods.

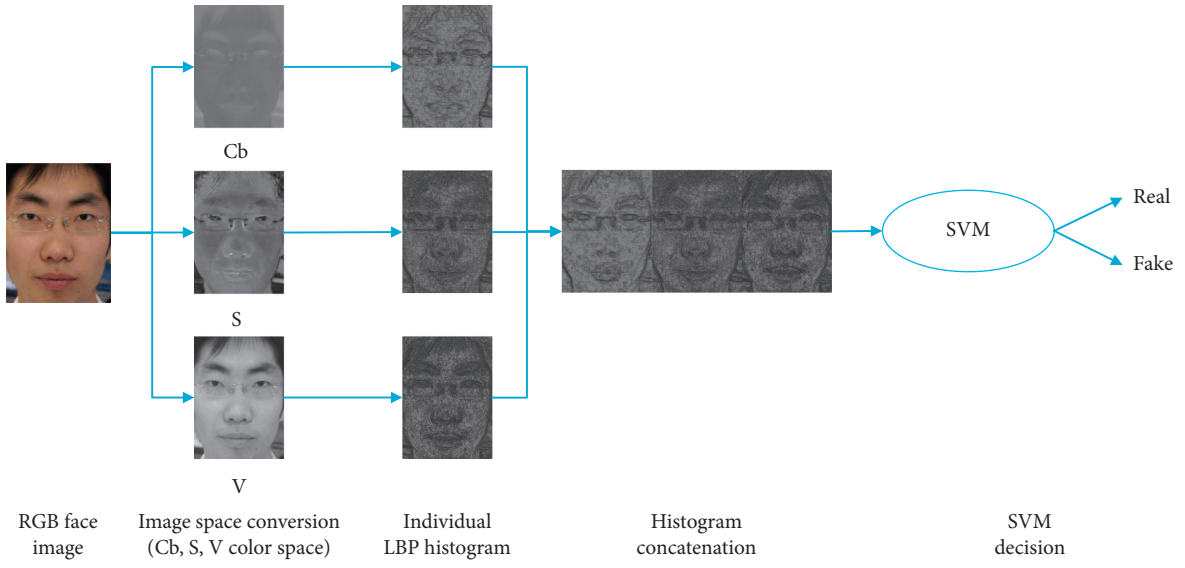

Figure 4: Conceptual diagram of the proposed scheme.

Table 1: Details on data partitioning in CASIA-FASD.

Type	Genuine images (ea)	Spoof images (ea)			
		Printed photo attacks	Cut photo attacks	Video replay attacks	Total
Training set	4,577	5,054	2,368	4,429	11,851
Testing set	5,912	7,450	4,437	5,652	17,539

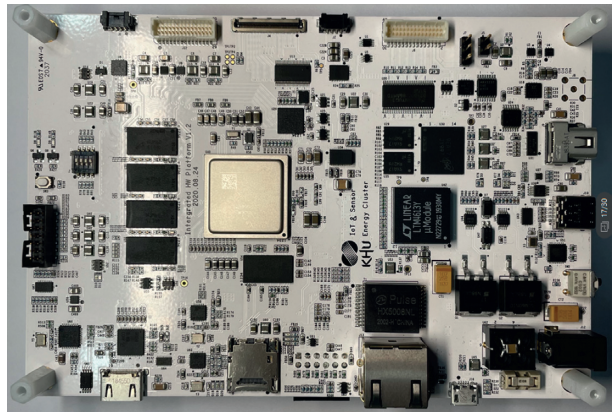

Figure 5: AI FPGA board.

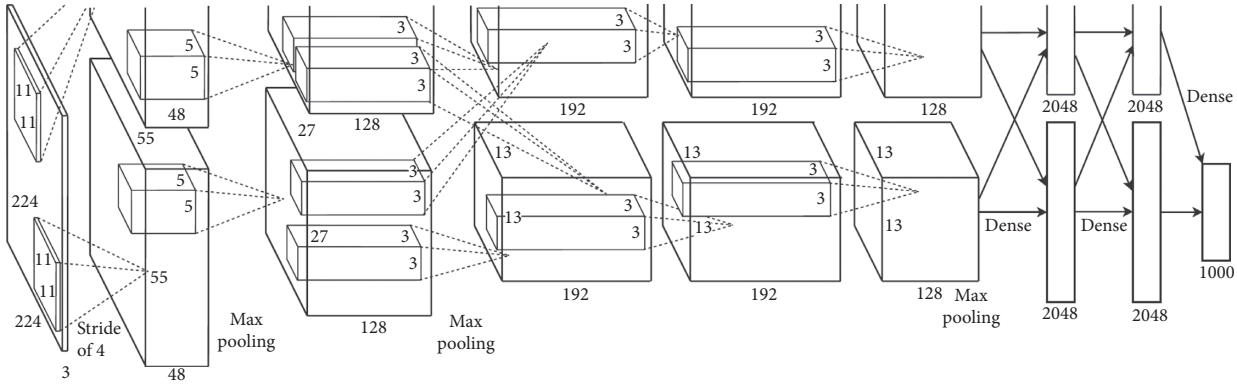

Figure 6: Illustration of the proposed CNN architecture, explicitly showing the delineation of responsibilities between the two GPUs: one GPU runs the layer parts at the top of the figure while the other runs the layer parts at the bottom.

The FRR [41] is a measure of how likely the biometric security system will incorrectly reject an access attempt by an authorized user. A system's FRR typically is defined as the ratio of the number of false rejections divided by the number of identification attempts.

Smaller HTER values indicate good performance, where HTER is defined using only misclassification ratios. Additionally, the EER (equal error rate) refers to the rate at which the FRR and FAR values converge to one another, where a small value also indicates good performance.

The EER [42] is a biometric security system algorithm used to predetermine the threshold values for the FAR and FRR. When the rates are equal, the common value is referred to as the equal error rate. The lower the ERR, the better the accuracy of the biometric system.

ROC (receiver operating characteristic) curve is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.

AUC (area under the curve) is the area under the ROC Curve. If the AUC value is high, it means that the model for classifying objects has excellent performance.

4.4. Experimental Results and Discussion. To verify the performance of the proposed scheme, eight scenarios were compared and tested using the CASIA-FASD attack dataset.

Table 2 shows HTERs according to eight different scenarios in the CASIA-FASD dataset. The proposed method showed improved performance for printed photo attacks, cut photo attacks, and video replay attacks. Figure 7 shows the performance comparison for the CASIA-FASD dataset.

Table 3 shows the EER values according to eight different scenarios for the CASIA-FASD dataset. Compared with the proposed scheme, only the "YCbCr_lbp + HSV_lbp" scheme has good EER performance.

The receiver operating characteristic (ROC) curves are presented. These curves show the error of the false positive rates against the true positive rates. ROC curves are best used for comparing the performance of various systems. Figures 8 and 9 show the ROC curves generated for each scenario in the CASIA-FASD dataset.

Table 4 shows the FAR, FRR, and area under the curve (AUC) results according to eight different scenarios in the CASIA-FASD dataset. A high AUC indicates good performance.

Table 5 shows the accuracy for different facial spoofing attacks. The accuracy for YCbCr_lbp + HSV_lbp is the

Table 2: Performance of various scenarios on the CASIA-FASD dataset.

Scenario	HTER (%)			
	Printed photo attacks	Cut photo attacks	Video replay attacks	Total
YCbCr	13.05	12.41	10.28	11.92
HSV	6.34	5.34	5.34	5.67
YCbCr_lbp	2.80	3.05	1.30	2.38
HSV_lbp	9.70	9.16	8.85	9.24
YCbCr + HSV	5.66	4.55	4.55	4.92
YCbCr_lbp + HSV	5.53	4.52	4.50	4.85
YCbCr_lbp + HSV_lbp	2.78	2.53	2.12	2.48
Proposed approach	**2.46**	**1.24**	**0.57**	**1.42**

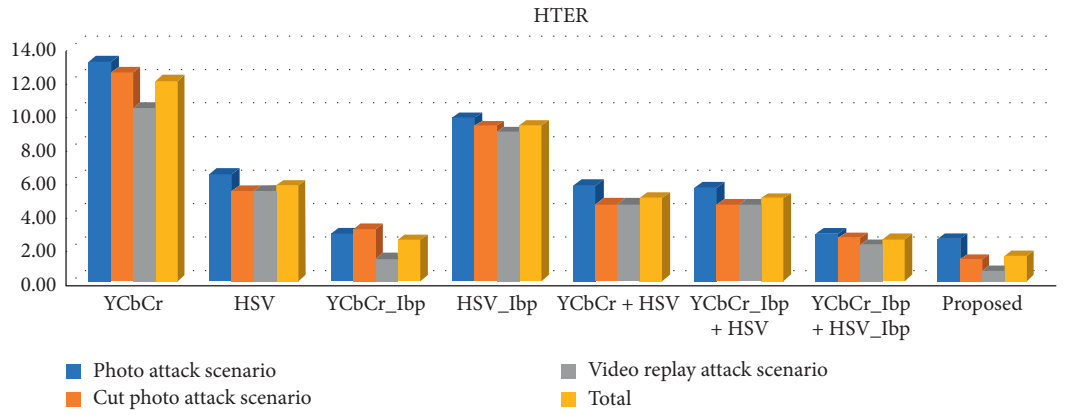

Figure 7: Performance comparison for the CASIA-FASD dataset.

Table 3: Equal error rate values for the CASIA-FASD dataset.

Scenario	EER (%)			
	Printed photo attacks	Cut photo attacks	Video replay attacks	Total
YCbCr	25.22	18.39	**5.39**	16.98
HSV	10.14	**0.00**	12.66	13.23
YCbCr_lbp	14.55	19.35	27.68	23.16
HSV_lbp	11.09	3.57	12.16	10.76
YCbCr + HSV	**6.13**	**0.00**	12.95	11.09
YCbCr_lbp + HSV	7.09	0.02	8.22	**7.58**
YCbCr_lbp + HSV_lbp	7.29	5.56	6.52	9.50
Proposed approach	10.79	12.91	7.76	10.22

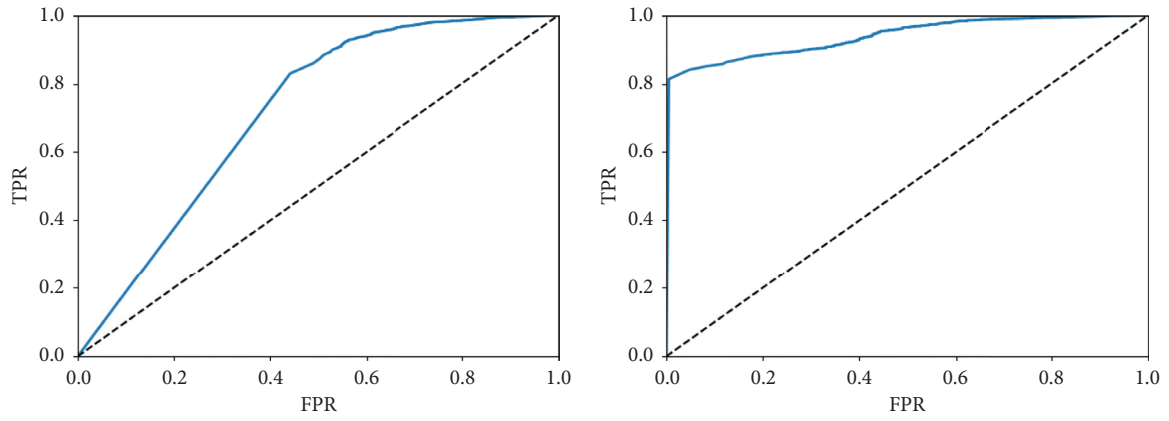

Figure 8: Continued.

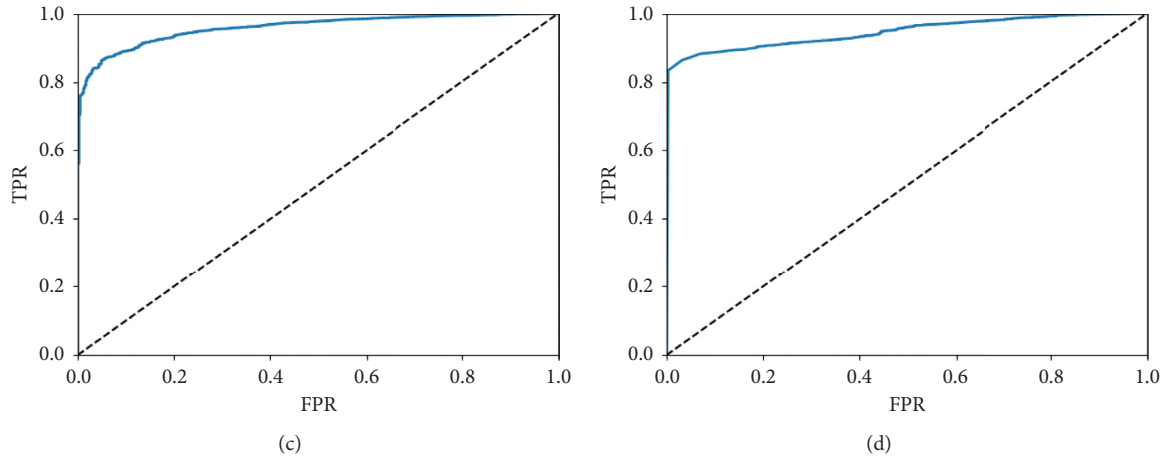

Figure 8: Receiver operating characteristic curves for the (a) YCbCr, (b) HSV, (c) YCbCr_lbp, and (d) HSV_lbp scenarios.

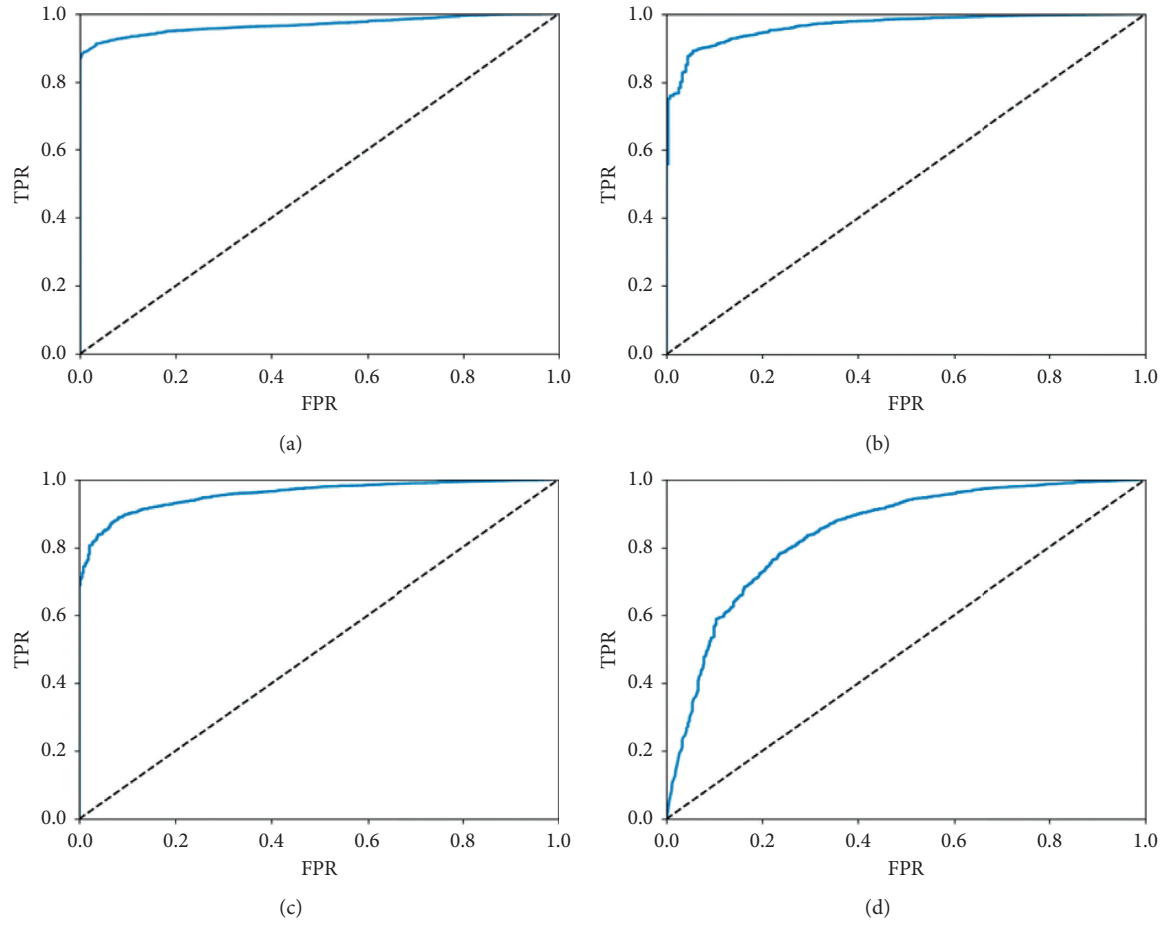

Figure 9: Receiver operating characteristic curves for the (a) YCbCr + HSV, (b) YCbCr_lbp + HSV, (c) YCbCr_lbp + HSV_lbp, and (d) proposed scenarios.

highest, but the proposed method shows similar performance.

The overall test results of this paper are shown in Table 6. Compared to the already existing YCbCr_lbp + HSV_lbp method, the method proposed in this paper has improved performance with respect to printed photo attacks (0.18%), cut photo attacks (0.69%), and video replay attacks (1.52%), with an overall performance improvement of 0.73%. Additionally, the ERR was low, while the accuracy values were similar. Overall, the YCbCr_lbp + HSV_lbp method showed

Table 4: FAR, FRR, and AUC performances for the eight scenarios.

Scenarios	FAR (%)				FRR (%)	AUC
	Printed photo attacks	Cut photo attacks	Video replay attacks	Total		
YCbCr	5.53	4.26	3.52	4.44	20.57	0.72
HSV	2.01	**0.00**	4.05	2.02	10.77	0.94
YCbCr_lbp	2.99	3.49	5.11	3.87	2.65	0.84
HSV_lbp	1.71	0.63	4.51	2.29	17.83	0.96
YCbCr + HSV	2.22	**0.00**	3.26	1.82	9.19	0.95
YCbCr_lbp + HSV	2.05	0.05	2.58	1.56	9.08	**0.97**
YCbCr_lbp + HSV_lbp	**1.30**	0.81	2.44	**1.52**	4.90	**0.97**
Proposed approach	3.78	1.35	**2.25**	2.46	**1.17**	0.96

Table 5: Accuracy comparison.

Scenarios	Accuracy (%)
YCbCr	91.34
HSV	95.66
YCbCr_lbp	96.49
HSV_lbp	93.73
YCbCr + HSV	96.19
YCbCr_lbp + HSV	96.42
YCbCr_lbp + HSV_lbp	**97.76**
Proposed approach	97.54

Table 6: Compare all results.

Scenarios	HTER (%)	EER (%)	FAR (%)	FRR (%)	AUC	Accuracy (%)
YCbCr	11.92	16.98	4.44	20.57	0.72	91.34
HSV	5.67	13.23	2.02	10.77	0.94	95.66
YCbCr_lbp	2.38	23.16	3.87	2.65	0.84	96.49
HSV_lbp	9.24	10.76	2.29	17.83	0.96	93.73
YCbCr + HSV	4.92	11.09	1.82	9.19	0.95	96.19
YCbCr_lbp + HSV	4.85	7.58	1.56	9.08	**0.97**	96.42
YCbCr_lbp + HSV_lbp	2.48	**9.50**	**1.52**	4.90	**0.97**	**97.76**
Proposed approach	**1.42**	10.22	2.46	**1.17**	0.96	97.54

similar performance but uses six color space channels, while the proposed method uses only three-color space channels, leading to a faster calculation speed.

5. Conclusions

In this paper, we proposed a face antispoofing method utilizing CNN learning and inference and constructed important parameters by extracting texture information via an LBP from the face image color space. CASIA-FASD was used as the dataset for performance verification. Images were extracted from videos and divided into printed photo attacks, cut photo attacks, and video replay attacks. These images extracted from the CASIA-FASD dataset were used for both training and evaluation. It was confirmed that the detection performance was improved by separating the color space from the face image in addition to the Cb, S, and V color space, which is useful for antispoofing. In previous studies, a 6-channel (YCbCr + HSV) color space was typically used, leading to large computational costs. On the contrary, the proposed approach reduces the computational load by instead considering only three (Cb, S, V) color space channels. Considering the AI FPGA board, the performances of the existing methods were evaluated with that of the proposed scheme. It was confirmed that the proposed method can be effectively used in edge environments.

As future work, we want to verify the performance against another well-known face spoof dataset. In addition, we plan to conduct performance tests between databases.

Data Availability

The data used to support the finding were included in this paper.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was funded by BK21 FOUR (Fostering Outstanding Universities for Research) (no. 5199990914048), and this research was supported by Basic Science Research Program through the National Research Foundation of

Korea (NRF) funded by the Ministry of Education (NRF-2020R1I1A3066543). In addition, this work was supported by the Soonchunhyang University Research Fund.

References

[1] Z. Akhtar and G. Luca Foresti, "Face spoof attack recognition using discriminative image patches," *Journal of Electrical and Computer Engineering*, vol. 2016, Article ID 4721849, 14 pages, 2016.

[2] H. K. Jee, S. U. Jung, and J. H. Yoo, "Liveness detection for embedded face recognition system," *International Journal of Biological and Medical Sciences*, vol. 1, pp. 235–238, 2006.

[3] W. Bao, H. Li, N. Li, and W. Jiang, "A liveness detection method for face recognition based on optical flow field," in *Proceedings of the 2009 International Conference on Image Analysis and Signal Processing IASP*, pp. 233–236, IEEE, Linhai, China, April 2009.

[4] J. Li, Y. Wang, T. Tan, and A. K. Jain, "Live face detection based on the analysis of fourier spectra," in *Proceedings of the SPIE - International Society for Optics and Photonics*, pp. 296–303, Choufu, Japan, March 2004.

[5] A. D. S. Pinto, H. Pedrini, W. R. Schwartz, and A. Rocha, "Video-based face spoofing detection through visual rhythm analysis," in *Proceedings of the 2012 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 221–228, IEEE, Ouro Preto, Brazil, August 2012.

[6] W. R. Schwartz, A. Rocha, and H. P. Edrini, "Face spoofing detection through partial least squares and low-level descriptors," in *Proceedings of the 2011 International Joint Conference on Biometrics (IJCB)*, pp. 1–8, IEEE, Washington, WA, USA, October 2011.

[7] A. Anjos and S. Marcel, "Counter-measures to photo attacks in face recognition: a public database and a baseline," in *Proceedings of the 2011 Joint Conference on Biometrics (IJCB)*, pp. 1–7, IEEE, Washington, WA, USA, October 2011.

[8] J. Määttä, A. Hadid, and M. Pietikainen, "Face spoofing detection from single images using micro-texture analysis," in *Proceedings of the 2011 international joint conference on Biometrics (IJCB)*, pp. 1–7, IEEE, Washington, WA, USA, October 2011.

[9] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[10] J. Yang, Z. Lei, and S. Z. Li, "Learn convolutional neural network for face anti-spoofing," 2014, https://arxiv.org/abs/1408.5601.

[11] O. Lucena, A. Junior, V. Moia, R. Souza, E. Valle, and R. Lotufo, "Transfer learning using convolutional neural networks for face anti-spoofing," in *Lecture Notes in Computer Science*Springer, Berlin, Germany, 2017.

[12] Z. Xu, S. Li, and W. Deng, "Learning temporal features using LSTM-CNN architecture for face anti-spoofing," in *Proceedings of 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 141–145, IEEE, Kuala Lumpur, Malaysia, November 2015.

[13] T. Pereira, A. Anjos, J. M. De Martino, and S´. Marcel, "Lbp-top based countermeasure against face spoofing attacks," in *Proceedings of Asian Conference on Computer Vision*, pp. 121–132, Springer, Daejeon, Korea, November 2012.

[14] K. Patel, H. Han, and A. K. Jain, "Secure face unlock: spoof detection on smartphones," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 10, pp. 2268–2283, 2016.

[15] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face anti-spoofing using speeded-up robust features and Fisher vector encoding," *IEEE Signal Processing Letters*, vol. 24, no. 2, pp. 141–145, 2017.

[16] J. Komulainen, A. Hadid, and M. Pietikainen, "Context based face anti-spoofing," in *Proceedings of the 2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pp. 1–8, Arlington, VA, USA, September 2013.

[17] P. Bruno, C. Michelassi, and R. Anderson, "Face liveness detection under bad illumination conditions," in *Proceedings of the 2011 18th IEEE International. Conference on Image Processing. (ICIP 2011)*, pp. 3557–3560, IEEE, Brussels, Belgium, September 2011.

[18] J. Komulainen, A. Hadid, and M. Pietik¨ainen, "Face spoofing detection using dynamic texture," in *Asian Conference on Computer Vision*, pp. 146–157, Springer, Daejeon, Korea, November 2012.

[19] T. Ahmad Siddiqui, S. Bharadwaj, T. I Dhamecha et al., "Face anti-spoofing with multifeature videolet aggregation," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 1035–1040, IEEE, Cancun, Mexico, December 2016.

[20] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1–8, Rio de Janeiro, Brazil, October 2007.

[21] L. Li, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, and A. Hadid, "An original face anti-spoofing approach using partial convolutional neural network," in *The sixth International Conference on Image Processing Theory, Tools and Applications (IPTA'16)*, pp. 1–6, Oulu, Finland, December 2016.

[22] K. Patel, H. Han, and A. K. Jain, "Cross-database face anti-spoofing with robust feature representation," in *Proceedings of the Chinese Conference on Biometric Recognition*, pp. 611–619, Springer, Chengdu, China, October 2016.

[23] A. George and S. Marcel, "Deep pixel-wise binary supervision for face presentation attack detection," in *Proceedings of the 2019 Inter-national Conference on Biometrics*, Crete, Greece, June 2019.

[24] J. Amin, Y. Liu, and X. Liu, "Face despoofing: anti-spoofing via noise modeling," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 290–306, Munich, Germany, September 2018.

[25] M. Sajid, N. Ali, S. Hanif Dar et al., "Data augmentation-assisted makeup-invariant face recognition," *Mathematical Problems in Engineering*, vol. 2018, Article ID 2850632, 10 pages, 2018.

[26] M. Alghaili, Z. Li, A. Hamdi, and R. Ali, "Face filter: face identification with deep learning and filter algorithm," *Scientific Programming*, vol. 2020, Article ID 7846264, 9 pages, 2020.

[27] Y. Xu, W. Yan, G. Yang et al., "Joint face detection and alignment using face as point," *Scientific Programming*, vol. 2020, Article ID 7845384, 8 pages, 2020.

[28] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face anti-spoofing based on color texture analysis," in *Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP)*, pp. 2636–2640, Quebec, Canada, September 2015.

[29] S. H. Lee, H. Kim, and Y. M. Ro, "A comparative study of color texture features for face analysis," in *Computational Color Imaging. CCIW*, S. Tominaga, R. Schettini, and A. Trémeau, Eds., Berlin, Heidelberg, Springer.

[30] G. Kim, S. Eum, J. Suhr, D. Kim, K. Park, and J. Kim, "Face liveness detection based on texture and frequency analyses,"

in *Proceedings of 2012 5th IAPR International Conference on Biometrics, ICB*, pp. 67–72, New Delhi, India, April 2012.

[31] G. Sang, L. Jing, and Q. Zhao, "Pose-invariant face recognition via RGB-D images," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 3563758, 9 pages, 2016.

[32] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions," in *Proceedings of the 12th IAPR International Conference on Pattern Recognition Conference A: Computer Vision & Image Processing,*, pp. 582–585, Jerusalem, Israel, October 1994.

[33] J. Galbally, S. Marcel, and J. Fierrez, "Biometric antispoofing methods: a survey in face recognition," *IEEE Access*, vol. 2, pp. 1530–1552, 2014.

[34] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[35] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li, "A face antispoofing database with diverse attacks," in *Proceedings of the 5th IAPR International Conference on Biometrics (ICB '12)*, pp. 26–31, IEEE, New Delhi, India, April 2012.

[36] https://www.xilinx.com/products/silicon-devices/soc/zynq-ultrascale-mpsoc.html.

[37] https://github.com/Xilinx/Vitis-AI.

[38] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 2012.

[39] Md R. Hasan, "Face anti-spoofing using texture-based techniques and filtering methods," in *2019 3rd International Conference on Machine Vision and Information Technology (CMVIT2019)*, Guangzhou, China, February 2019.

[40] https://www.webopedia.com/definitions/false-acceptance.

[41] https://www.webopedia.com/definitions/false-rejection/.

[42] https://www.webopedia.com/definitions/equal-error-rate/.