



A novel normalization algorithm to facilitate pre-assessment of Covid-19 disease by improving accuracy of CNN and its FPGA implementation

Sertaç Yaman¹ · Barış Karakaya² · Yavuz Erol²

Received: 24 November 2021 / Accepted: 18 January 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract

COVID-19 is still a fatal disease, which has threatened all people by affecting the human lungs. Chest X-Ray or computed tomography imaging is commonly used to make a fast and reliable medical investigation to detect the COVID-19 virus. These medical images are remarkably challenging because it is a full-time job and prone to human errors. In this paper, a new normalization algorithm that consists of Mean–Variance–Softmax–Rescale (MVSR) processes respectively is proposed to provide facilitation pre-assessment and diagnosis Covid-19 disease. In order to show the effect of MVSR normalization technique, the algorithm of proposed method is applied to chest X-ray and Sars-Cov-2 computed tomography images dataset. The normalized X-ray images with MVSR are used to recognize Covid-19 virus via Convolutional Neural Network (CNN) model. At the implementation stage, the MVSR algorithm is executed on MATLAB environment, then all the arithmetic operations of the MVSR normalization are coded in VHDL with the help of fixed-point fractional number representation format on FPGA platform. The experimental platform consists of Zynq-7000 Development FPGA Board and VGA monitor to display the both original and MVSR normalized chest X-ray images. The CNN model is constructed and executed using Anaconda Navigator interface with python language. Based on the results of this study, infections of Covid-19 disease can be easily diagnosed with MVSR normalization technique. The proposed MVSR normalization technique increased the classification accuracy of the CNN model from 83.01, to 96.16% for binary class of chest X-ray images.

Keywords Normalization · FPGA · Covid-19 disease · Deep Learning · Image processing

1 Introduction

Normalization with mean and variance is a noise/error compensation method that is used in many applications such as signal, image and speech processing, artificial intelligence, big data, optimization algorithms and so on (Scalco et al. 2020; Song et al. 2020; Xu et al. 2020). One of the

compensation techniques is Batch normalization (BN) method to develop performance and stability of neural networks by rescaling values of hidden layer units and also to create more complicated deep learning models. One of the critical problems is the covariate shift in the learning process for deep learning models. To solve this problem, BN is proposed in 2015 (Ioffe and Szegedy 2015). Besides, Cepstral mean and variance normalization (CMVN) (Joshi et al. 2016) is used to eliminate noise inside voice for speech processing applications. CMVN is a critical method for recognizing voice instructions to minimize the data lost for messages during normalization.

Noise in the image is a general problem for image processing applications (Hong et al. 2020). It is also hot topic in computer vision and image processing. Many techniques based on image de-noising model have been developed to overcome this problem (Islam et al. 2018; Huber et al. 2021; Noor et al. 2020). In most studies, researchers firstly add noise (especially Gaussian noise) with different noise

✉ Sertaç Yaman
sertacyaman@hakkari.edu.tr

Barış Karakaya
bkarakaya@firat.edu.tr

Yavuz Erol
yerol@firat.edu.tr

¹ Faculty of Engineering, Department of Electrical-Electronics Engineering, Hakkari University, 30000 Hakkari, Turkey

² Faculty of Engineering, Department of Electrical-Electronics Engineering, Firat University, 23119 Elazığ, Turkey

ratios to the original image, then they design deep CNNs to enhance image de-noising performance (speed, quality and cost) and efficiency (Zhang et al. 2017). Some of these techniques consists of CNN with BN (Tian et al. 2020), image restoration by applying ResNet (Ren et al. 2019), several traditional linear and non-linear image filter techniques (Motwani et al. 2004).

Deep Normalized Convolutional Neural Network (DNCNN) is developed for unbalanced classifications of neural networks and constructed based on ReLU and weight normalization strategy. In (Jia et al. 2018), ReLU is employed as an activation function to overcome the gradient vanishing problem and enhance the optimization by using the parameters (weight, bias) normalization strategy (Nair and Hinton 2010). Weight Normalization (WN) is used to accelerate the convergence of stochastic gradient descent optimization in a neural network. In Salimans and Kingma (2016), WN is applied to show benefits of this technique in four different NN models. WN method is also used for image smoke detection in Yin et al. (2017). The result of this study shows that false alarm ratio is achieved under 0.60% and smoke detection accuracy is around 96.37%.

CNN models have been constructed in many different ways (Saponara et al. 2021; Supraja et al. 2021) for image processing (such as image segmentation, classification and de-noising), but still there are many challenge during image processing. These models require amounts of parameters (image size, weights, and bias, etc.) and lots of convolution operations in the training and inference process (Köpüklü et al. 2019; Nayak et al. 2020; Yamashita et al. 2018; Agrawal and Choudhary 2021). This has inspired many researchers to focus their attention on more hardware platforms with energy-efficiency (Duan et al. 2018; Pang et al. 2020).

Field Programmable Gate Array (FPGA) is a vital side for implementation of CNN models due to its hardware adaptability and parallel process capability. On the other hand, General Processing Units (GPUs) supply more resources in terms of memory and computational units. But, during the validation process, GPUs have a long execution time because of sequential logic design (Wang et al. 1999; Zhijie et al. 2020). FPGA has parallel architectures on its logic configuration. This feature makes it better choice for hardware designers because of the validation process of CNN configurations. At the implementation stage of CNN models, normalization process ensures fewer number of hardware resources and reduced power consumption during the training and validation processes on any hardware platforms (Wang et al. 1999; Zhijie et al. 2020; Nurvitadhi et al. 2016; Baptista et al. 2020).

COVID-19 is a pandemic disease, and it can threaten the health of all people in the world. It directly attacks the lung cells, and if not diagnosed early, can cause damage,

including mortality. Therefore, in this paper a new normalization algorithm that consists of MVSR (named as acronym of Mean–Variance–Softmax–Rescale) processes respectively is proposed. Mean and Variance techniques have generally been used for noise recompense in signal processing (especially speech) applications to eliminate the mismatch between training and test signals by changing to zero mean and unit variance (Subramanian et al. 2021). In order to show effect of MVSR on image processing, the proposed algorithm is applied to chest X-ray and CT images to provide facilitation pre-assessment and diagnosis Covid-19 disease. Therefore, the normalized X-ray images with MVSR are used to recognize Covid-19 virus via one of the neural network models as known Convolutional Neural Networks (CNNs). At the implementation stage, the MVSR algorithm is executed on MATLAB, then it is implemented on FPGA platform. The experimental platform consists of Zynq-7000 Development Board and VGA monitor to display the both original X-ray and MVSR normalized image. The CNN model is constructed and executed using Anaconda Navigator interface with python language.

The rest of the paper is organized as follows. The architecture, mathematic of the MVSR normalization technique and its necessity are detailed in Sect. 2. In Sect. 3, the MATLAB pre-study is accomplished for observing the effect of proposed normalization algorithm on chest X-ray images. Section 4 consists of FPGA implementation of MVSR normalization technique and the experimental results for chest X-ray images on VGA screen. Then, in Sect. 5, the architecture of CNN model is constructed and the implementation of image classification (diagnosis of Covid-19 disease) for original and normalized chest X-ray images are executed on Anaconda Navigator interface with python language to prove the benefits of proposed MVSR normalization algorithm. Section 6 concludes the paper.

2 MVSR Normalization Algorithm

In general, the standard deviation (the square root of the variance) is used to make a relation between each data and measure how to spread out or distribute around the mean of a data set (Singh and Singh 2020). The MVSR normalization is built on these features of mathematical operations, which are include four-part processes as follows;

- Mean
- Variance
- Softmax
- Rescaling of the data, respectively.

According to R. Duncan Luce's probability theory, it is also known as Luce's choice axiom, probability of one

of the sample is dependent on another sample in the same data set. The softmax function, based on Luce's choice axiom, is usually employed in the last activation function of multiclass Artificial Neural Network (ANN) models to make a probability distribution of the output classes (Angelov and Soares 2020). After calculation of the normalized intensity of the input, the dataset can have the negative and the positive fractional numbers. In order to conserve the effect of negative data and also nonlinearity, we used the softmax function. it is one of the main contributions to the state of the art this paper proposes. In the rescaling step, all data is transformed to 8-bit unsigned integer intervals change from 0–1 to 0–255. The MVSR normalization technique consists of following mathematical processes where $\mu_{MVSR}, \sigma_{MVSR}^2, \hat{x}_i, S(x)_i$ and r_i stand for mean of x_i , variance, the normalized intensity of the input, softmax and rescaled values, respectively.

$$\mu_{MVSR} = \frac{1}{M} \sum_{i=1}^M x_i \tag{1}$$

$$\sigma_{MVSR}^2 = \frac{1}{M} \sum_{i=1}^M (x_i - \mu_{MVSR})^2 \tag{2}$$

$$\hat{x}_i = \frac{(x_i - \mu_{MVSR})}{\sqrt{\sigma_{MVSR}^2 + \delta}} \tag{3}$$

$$S(x)_i = \frac{e^{\hat{x}_i}}{\sum_j e^{\hat{x}_j}} \tag{4}$$

$$r_i = \frac{S(x)_i}{\max(S(x)_i)} * 255 \tag{5}$$

In the normalized intensity value of the input (\hat{x}_i), a small positive number (δ) is added to the variance value

to avoid the divisor is to be zero. In the equations above, x_i is the input data intensity and r_i is the result of MVSR normalization technique.

The MVSR normalization architecture is given in Fig. 1. The mathematical operations are executed in order to represented flow. The chest X-ray images are used in the MVSR algorithm as input variables and later, the normalized input images are applied to the CNN model to make a classification.

In the pre-study, the proposed normalization algorithm is executed on MATLAB program to observe the effect of MVSR normalization for Covid-19 disease diagnosis from chest X-ray and CT images. Chest X-ray images of Covid-19 infected and No-Finding are represented after and before MVSR normalization in the Fig. 2. The sources, chest X-ray and CT images datasets, which has been generally used in the literature, are publicly available from Hospital Israelita Albert Einstein (Sertaç et al. 2019; Kingma and Ba 1412). According to the expert view, Infectious Diseases Doctor, the infection is more clearly observed after applying the MVSR normalization techniques on chest X-ray images. At this stage, the starting point of this study is based on this view and this normalization process is thought to be used in image classification applications. The accuracy of Covid-19 diagnose is increased by using the proposed MVSR normalization technique before training process for the neural network models. The rest of the paper is based on this projection.

The proposed MVSR normalization technique is applied to both the Chest X-ray and the SARS-CoV-2 CT computed tomography datasets and the results obtained are shown in Figs. 2 and 3. When comparing the normalized image with the original image, it is clearly seen that the proposed normalization increases the quality of the images and facilitates the detection of infected regions.

MVSR normalization technique used on the data sets via the MATLAB program for the pre-phase. This process was taken as a reference to design the FPGA-based MVSR normalization technique. In order to show benefit of the MVSR

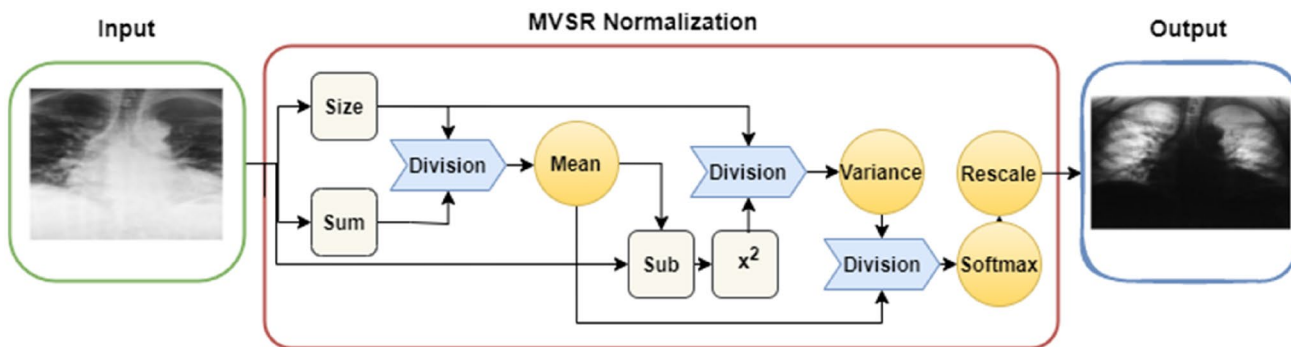
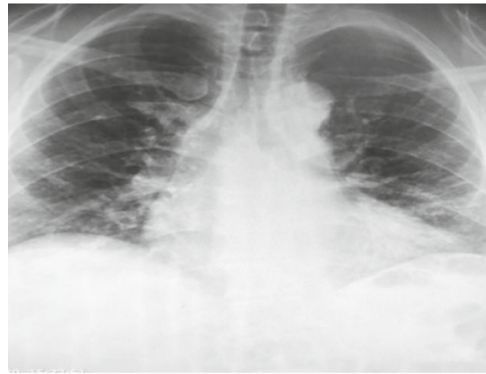
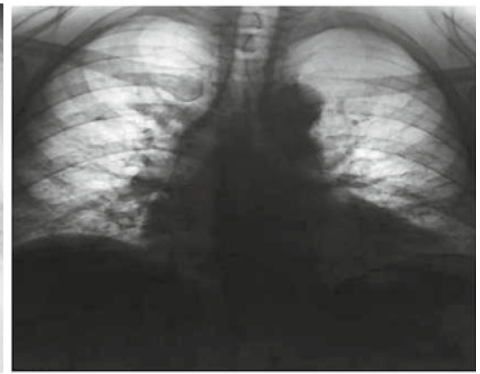


Fig. 1 MVSR architecture

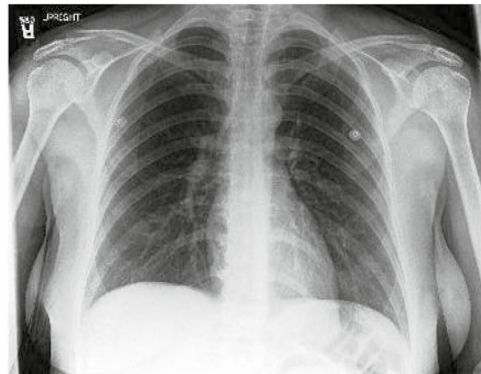
Fig. 2 **a, c** Are original chest X-ray images while **b, d** are the normalized versions of **a, c**



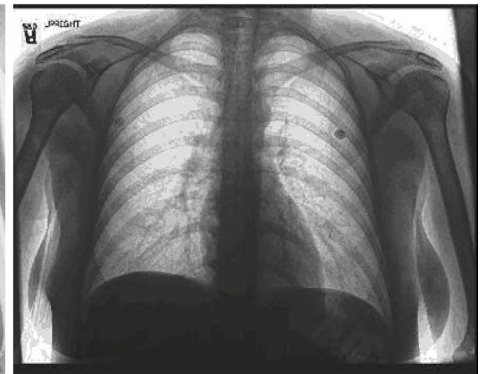
a. Covid-19 X-ray



b. Covid-19 MVSR

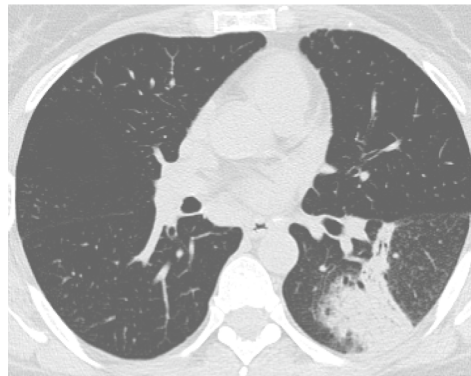


c. No-Finding X-ray

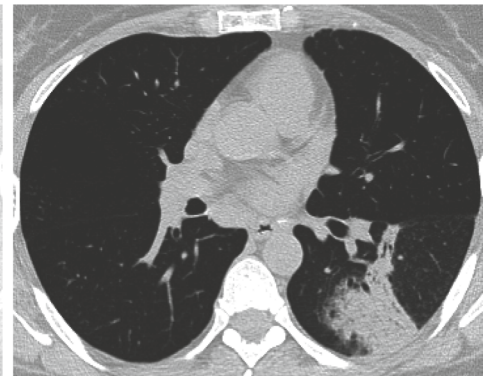


d. No-Finding MVSR

Fig. 3 **a, c** Are original Sars-Cov-2 scan images while **b, d** are the normalized versions of **a, c**



a. Covid-19 Sars-Cov-2



b. Covid-19 MVS



c. No-Finding Sars-Cov-2



d. No-Finding MVS

normalization, more images are normalized and presented in Fig. 4.

3 MVSR Normalization on FPGA

In order to apply the proposed MVSR normalization algorithm, the operations, standard deviation, mean, and variance are required to calculate in sequence. To obtain these operations, we need to define real numbers as a fixed point number type for embedded systems or any hardware languages. Multiplication, division, addition, subtraction, and especially square root calculations are essential mathematical arithmetic operations to execute proposed MVSR normalization technique on FPGA platform. MVSR normalization is implemented by using a static random access memory (SRAM) on re-programmable FPGA. All the data is read from SRAM and defined arrays. Systolic arrays are used for these operations to perform parallel integrations such as convolution, correlation, linear algebra or additional data tasks (Angelov and Soares 2020). All normalization processes are constructed with pure VHDL language at Vivado 2018.1 version.

3.1 Division of the fixed-point fractional binary numbers

Division is one of the important operation to calculate standard deviation, mean, and variance. Fixed-point fractional number format is a way to represent fractional numbers. The fixed point division process is slightly different from the standard binary division operation. It is needed to scale the fractional part of the division after finding the remainder (Schwab et al. 2020).

Algorithm 1. Pseudo Code for Division Operation

```

If Dividend < 0
    Dividend = Not Dividend +1
else if Divisor < 0
    Divisor = Not Divisor +1
else if Divisor or Dividend = 0
    Quotient=0
    Remainder=0
else
    Remainder = Dividend rem Divisor
    Quotient= Dividend / Divisor
    nRemainder=100* Remainder/ Divisor
    mRemainder = LUT_operation (nRemainder)
End
Output= Quotient& mRemainder

```

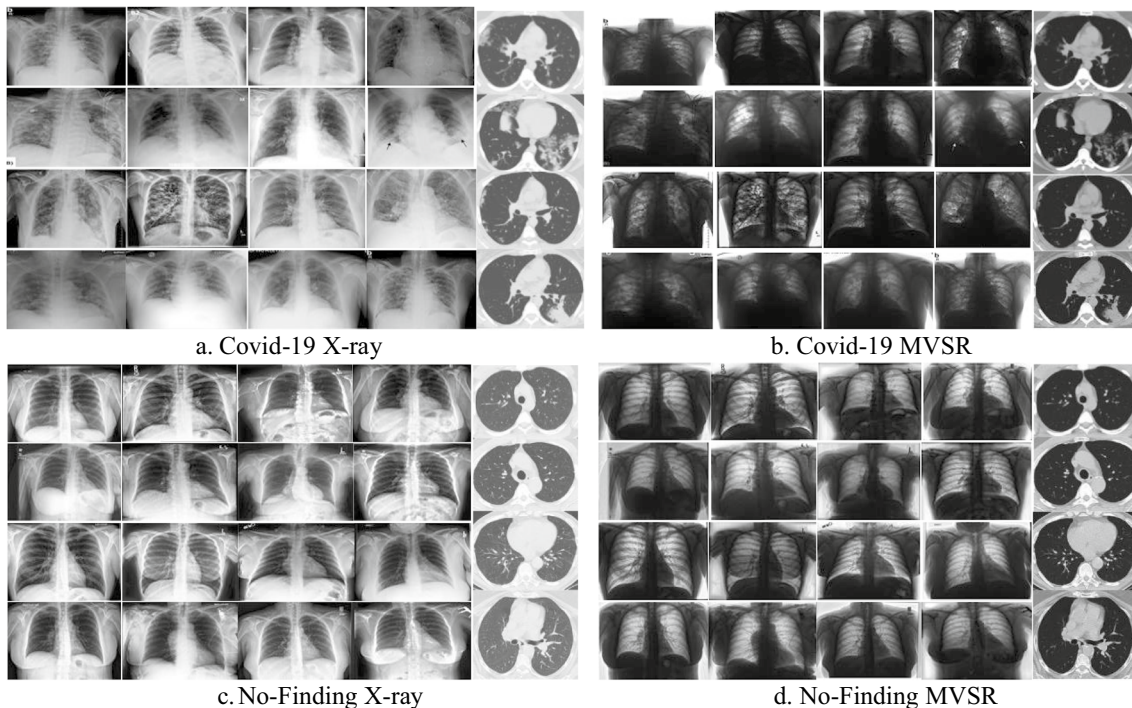


Fig. 4 a, c Are original chest X-ray images while b, d are the normalized versions of a, c

The *LUT_operation* consists of lookup table to provide programmable hardware functionality for division operation and to validate remainder values by matching against a list of valid items in the array. The values stored in LUT are in the unsigned fractional fixed point binary numbers. For example, when the *nReminder* is equal to "00,010,110" (stands for 0.22 in decimal), *mRemainder* is "00,111,000" in 8-bit format.

3.2 Square root of the fixed-point fractional binary numbers

This section represents calculating the fixed-point square root. Square root of the fixed point binary numbers is essential to have normalized input data with standard deviation from variance operation in the MVSR operation. After making the groups of data by 2^n ($n = 1, 2 \dots m, m = \text{data size}/2$), it is gradually compared with the multiplied expected data. If the group of the data is smaller than expected, the result is equal to 1, vice versa 0.

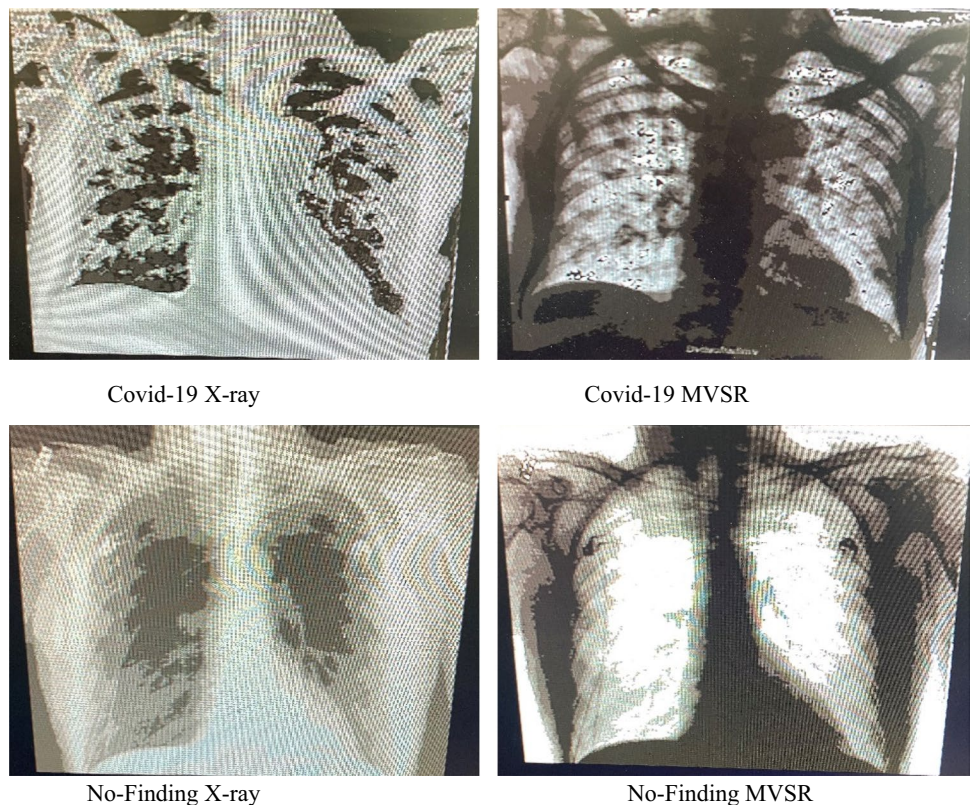
Algorithm 2. Square root operation on VHDL

```

datau:=unsigned(datain);
expectdata(1 downto 0):="01";
for i in 0 to length loop
  if (datau(size downto size-1-2*i)< expectdata(2*i+1 downto 0)) then
    dtu(length-i):='0';
  else
    dtu(length-i):='1';
  end if;
  expectdata(2*i+3 downto 0) :=(dtu(length downto length-i)&'1')*(dtu(length downto length-i)&'1');
  dt:=std_logic_vector(dtu);
end loop;
return dt;

```

Fig. 5 FPGA implementation of MVSR normalization and the results for chest X-ray images on VGA screen



The FPGA implementation platform where Xilinx Zynq-7000 Development Board is used to implement the MVSR normalization algorithm for input images and VGA monitor to display the both original and MVSR normalized image.

Results of FPGA implementation of MVSR normalization algorithm is given in Fig. 5. As seen in this figure, the algorithm is implemented on FPGA board and the results are shown on VGA monitor successfully. The main result of FPGA implementation stage is that the X-ray image can be normalized on FPGA and the expert views support our opinions on the starting point of this study.

In the Table 1, information about resource utilization of FPGA implementation stage of the present study is shown. This result shows that the MVSR normalization algorithm is performed on FPGA chip with the fewer number of resource.

4 Proposed CNN architecture and results obtained

The architecture of the CNN model is constructed with seven convolutional layers and five max-pooling layers and this model can be seen in Table 2. This model is executed for original and MVSR normalized chest X-ray images on Anaconda Navigator interface with python language to prove the benefits of the proposed MVSR normalization algorithm.

There are four-type of prediction approaches, four distinct combinations of predicted and actual values, to evaluate results of classification model via confusion matrix (Zeng 2020; Wu et al. 2019). These are True Positives (T_p), True Negatives (T_N), False Positives (F_p) and False Negatives (F_N).

- T_p : These are the people who are diagnosed with Covid-19 infection by clinical tests and the CNN model validates them as Covid-19 infected.
- T_N : These are the people who are not diagnosed with Covid-19 disease by clinical tests and the CNN model validates them as not infected.
- F_p : These are the people who are not diagnosed with Covid-19 disease by clinical tests but the CNN model classified them as Covid-19 infected.

Table 1 Resource utilization of FPGA Chip

Resource	Utilization	Available	Utilization %
LUT	193	53,200	0.36
FF	136	106,400	0.13
BRAM	30	140	21.43
IO	18	200	9

Table 2 CNN architecture

Layers	Layer Configurations			
	S	N	W	H
Input	–	–	150	150
Conv1	3×3	32	150	150
Conv2	3×3	32	150	150
Conv3	3×3	64	150	150
Max-Pool	2×2	–	75	75
Conv4	3×3	128	75	75
Max-Pool	2×2	–	37	37
Conv5	3×3	64	37	37
Max-Pool	2×2	–	18	18
Conv6	3×3	128	18	18
Max-Pool	2×2	–	9	9
Conv7	3×3	128	9	9
Max-Pool	2×2	–	4	4
FC1		2048		
FC2		512		
Output		1		

- F_N : These are the people who are diagnosed with Covid-19 disease by clinical tests but the CNN model classified them as not infected.

F1-score is the harmonic mean of the *Precision* and *Recall*. Therefore, f1-score gives a combined idea of these two measurements. F1-score is maximum when the *Precision* is equal to *Recall*. The precision determines whether the model is reliable or not. Prediction according to the model, the recall is given how many true positive cases are accurately predicted.

Precision; It counts how precise the executed model operates by examining TP from the predicted ones

$$P = \frac{T_p}{(T_p + F_p)} \tag{6}$$

Recall (Sensitivity); It is the percentage of recognition of positive samples

$$R = \frac{T_p}{(T_p + F_N)} \tag{7}$$

Table 3 Obtained F1-scores

	F1-score	F1-score MVSR
Covid-19	0.51	0.92
No-finding	0.85	0.96

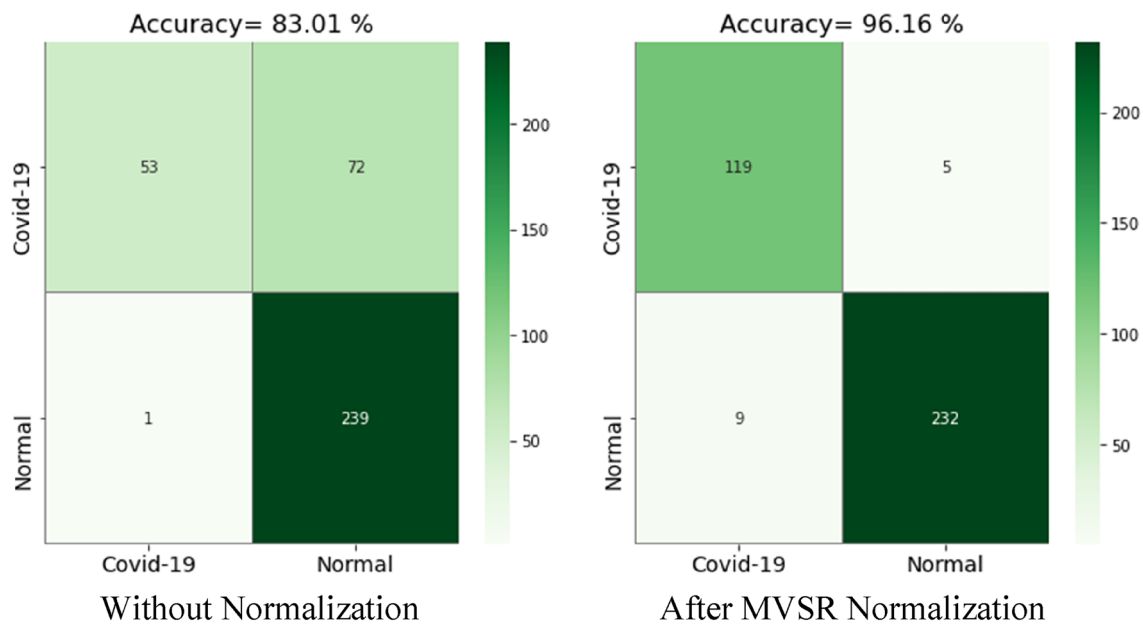


Fig. 6 Confusion Matrix

F-1 Score; It is a function to find a balance between sensitivity and precision.

$$F1 = \frac{2}{\left(\frac{1}{R} + \frac{1}{P}\right)} \quad (8)$$

The results of the f1-scores given in Table 3, performed with original chest X-ray and MVSR normalized images, the normalized images perform the better F1-Score (accuracy).

Figure 6 shows the confusion matrix of models, which are trained with original and MVSR normalized images. This matrix demonstrate how the CNN models classify and misclassify Covid-19 disease on chest X-ray images. Furthermore, the precision, recall and f1-score can be easily computable by using confusion matrix.

Table 4 briefly compares our proposed model with the other studies in terms of accuracy, which is found from recall and precision. Although the studies in the literature use transfer-learning, many pre-processing algorithms and traditional big machine learning models, the proposed model has higher classification accuracy by using the simple model and low execution time with less learning parameters. Before normalization, the other studies can have better results but when applying the MVSR normalization to Chest and CT images, the results of our model are better than in the studies of the literature, which are presented in Table 4.

5 Conclusion

In this paper, we propose a novel method, the MVSR normalization architecture, to facilitate pre-assessment of Covid-19 disease from Chest X-ray and CT images. The datasets are contained 2482 CT scan slices and 465 Chest X-ray images. The effectiveness of this new normalization technique is proved by using different evaluation methods, which include expert view, MATLAB, Anaconda Navigator interface and all the arithmetic operations of the MVSR normalization are coded in VHDL language with the help of the fixed-point fractional number representation format for FPGA embedded system. The proposed MVSR normalization technique increase the accuracy of the CNN model from 83.01, to 96.16% on chest X-ray images. In summary, the proposed normalization algorithm has the following benefits:

- (i) MVSR normalization considerably improves the accuracy of the CNN model and according to expert view, the difference of Covid-19 infected and uninfected CT and X-ray images can be seen more clearly at the pre-assessment stage.
- (ii) MVSR normalization can be applied to complex images like CT and chest X-ray images for recognizing the difference between infected and uninfected patients with Covid-19.
- (iii) Covid-19 disease can be efficiently diagnosed for MVSR normalized image.

Table 4 Normalization techniques used for the detection of COVID-19 patients in literature

Preprocessing	DL technique	Number of cases	Performance accuracy (%)		Study
			Without norm	With norm	
Segmentation, rescaling, multi-view fusion	ResNet50	368 COVID-19 Patients, 127 Patients with Other Pneumonia	NA	76	Wu et al. (2019)
Different features, early fusion, late fusion, different resampling algorithms	Inception-V3	90 COVID-19, 10 MERS, 11 SARS, 10 Varicella, 12 Streptococcus, 11 Pneumocystis Samples	83	88	Pereira et al. (2020)
GAN, rescaling, cropping	DECAPS architecture	349 COVID-19 Images, 397 Non-COVID-19 Images	84.3	87.6	Mobiny et al. (2004)
Rescaling	COVIDX-Net (VGG19, DenseNet201)	25 COVID-19, 25 Normal Cases	83	90	Hemdan et al. (2003)
Lung X-ray aperture in images region removed, histogram equalization and bilateral filter	VGG16, transfer learning with ImageNet	415 covid-19 infected, 5179 other to viruses having and 2880 normal	88	94.5	Heidari et al. (2020)
CXI image processing techniques	GoogleNet M-inception, transfer learning with ImageNet	1065 images, Covid-19 and Normal CXI	82.5	93	Rahman et al. (2021)
Histogram equalization, negative conversion, gamma correction, contrast enhancement techniques	ResNet18, ResNet50, ResNet101, InceptionV3, DenseNet201, ChexNet	18,479 images, 3616 Covid-19	93.22	Av. 96.29	Shoeibi et al. (2007)
MVSR normalization	CNN	240 covid-19, 120 normal, 200 test	83.01	96.16	This study

Main contributions to the state of the art this paper proposes,

- (i) The proposed normalization algorithm enhances the accuracy of the CNN model. Infected and uninfected regions can be seen more clearly on CT and X-ray images.
- (ii) The softmax function is to be the conserved effect of negative values and nonlinearity after applying mean and variance operations on every image.
- (iii) The proposed algorithm is implemented on the FPGA platform. The covid-19 virus can be diagnosed on an independent embedded platform by using this experimental platform.

Acknowledgements The authors would like to thank Prof. Dr. Kutbeddin DEMİRDAĞ for his valuable expert view, discussions and suggestions.

References

- Agrawal T, Choudhary P (2021) FocusCovid: automated COVID-19 detection using deep learning with chest X-ray images. *Evolv Syst* 2021:1–15
- Angelov P, Soares EA (2020) SARS-CoV-2 CT-scan dataset: a large dataset of real patients CT scans for SARS-CoV-2 identification. *MedRxiv*
- Baptista D, Sousa L, Morgado-Dias F (2020) Raising the abstraction level of a deep learning design on FPGAs. *IEEE Access* 8:205148–205161
- “Covid-19: covid-19 x-ray images”, GitHub, available online, <https://github.com/search?q=covid-19+x-ray+images>
- Duan Y et al (2018) Energy-efficient architecture for FPGA-based deep convolutional neural networks with binary weights. In: 2018 IEEE 23rd international conference on digital signal processing (DSP). IEEE
- Heidari M et al (2020) Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with pre-processing algorithms. *Int J Med Inform* 144:104284
- Hemdan EE, Shouman MA, Karar ME (2020) Covidx-net: a framework of deep learning classifiers to diagnose covid-19 in X-ray images. *arXiv preprint arXiv: 2003.11055*
- Hong JH et al (2020) Incremental image noise reduction in coronary CT angiography using a deep learning-based technique with iterative reconstruction. *Korean J Radiol* 21(10):1165
- Huber NR et al (2021) Random search as a neural network optimization strategy for Convolutional-Neural-Network (CNN)-based noise reduction in CT. *Medical Imaging 2021: Image Processing*, vol 11596. International Society for Optics and Photonics
- Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. PMLR
- Islam MT et al (2018) Mixed Gaussian-impulse noise reduction from images using convolutional neural network. *Signal Process Image Commun* 68:26–41
- Jia F et al (2018) Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization. *Mech Syst Signal Process* 110:349–367
- Joshi V, Prasad NV, Umesh S (2016) Modified mean and variance normalization: transforming to utterance-specific estimates. *Circ Syst Signal Process* 35(5):1593–1609
- Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. *arXiv preprint arXiv: 1412.6980*
- Köpüklü O et al (2019) Real-time hand gesture detection and classification using convolutional neural networks. In: 2019 14th IEEE international conference on automatic face & gesture recognition (FG 2019). IEEE
- Mobiny A et al (2020) Radiologist-level covid-19 detection using ct scans with detail-oriented capsule networks. *arXiv preprint arXiv: 2004.07407*
- Motwani MC et al (2004) Survey of image denoising techniques. In: *Proceedings of GSPX*, vol 27
- Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines. *Icml*
- Nayak DR, Dash R, Majhi B (2020) Automated diagnosis of multi-class brain abnormalities using MRI images: a deep convolutional neural network based method. *Pattern Recogn Lett* 138:385–391
- Noor A et al (2020) Median filters combined with denoising convolutional neural network for Gaussian and impulse noises. *Multimedia Tools Appl* 79(25):18553–18568
- Nurvitadhi E et al (2016) Accelerating binarized neural networks: comparison of FPGA, CPU, GPU, and ASIC. In: 2016 International conference on field-programmable technology (FPT). IEEE
- Pang W, Chenglu Wu, Shengli Lu (2020) An energy-efficient implementation of group pruned CNNs on FPGA. *IEEE Access* 8:217033–217044
- Pereira RM et al (2020) COVID-19 identification in chest X-ray images on flat and hierarchical classification scenarios. *Comput Methods Progr Biomed* 194:105532
- Rahman T et al (2021) Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images. *Comput Biol Med* 132:104319
- Ren D et al (2019) Progressive image deraining networks: a better and simpler baseline. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*
- Salimans T, Kingma DP (2016) Weight normalization: a simple reparameterization to accelerate training of deep neural networks. *arXiv preprint arXiv: 1602.07868*
- Saponara S, Elhanashi A, Gagliardi A (2021) Real-time video fire/smoke detection based on CNN in antifire surveillance systems. *J Real-Time Image Proc* 18(3):889–900
- Scalco E et al (2020) T2w-MRI signal normalization affects radiomics features reproducibility. *Med Phys* 47(4):1680–1691
- Schwab P et al (2020) Clinical predictive models for COVID-19: systematic study. *J Med Internet Res* 22(10):e21439
- Sertaç Y, Karakaya B, Erol Y (2019) Real time edge detection via IP-core based sobel filter on FPGA. In: 2019 International conference on applied automation and industrial diagnostics (ICAAID), vol 1. IEEE
- Shoeibi A et al (2020) Automated detection and forecasting of covid-19 using deep learning techniques: a review. *arXiv preprint arXiv: 2007.10785*
- Singh D, Singh B (2020) Investigating the impact of data normalization on classification performance. *Appl Soft Comput* 97:105524
- Song G et al (2020) Deep convolutional neural network based medical concept normalization. *IEEE Trans Big Data* 1:1
- Subramanian AS et al (2021) Directional ASR: a new paradigm for E2E multi-speaker speech recognition with source localization. In: *ICASSP 2021–2021 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE
- Supraja P et al (2021) 3D convolution neural network-based person identification using gait cycles. *Evolv Syst* 12(4):1045–1056
- Tian C, Yong Xu, Zuo W (2020) Image denoising using deep CNN with batch renormalization. *Neural Netw* 121:461–473

- Wang Yu, Chen Q, Zhang B (1999) Image enhancement based on equal area dualistic sub-image histogram equalization method. *IEEE Trans Consum Electron* 45(1):68–75
- Wang S et al (2021) A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19). *Eur Radiol* 31:1–9
- Wei X et al (2017) Automated systolic array architecture synthesis for high throughput CNN inference on FPGAs. In: *Proceedings of the 54th annual design automation conference 2017*
- Wu X et al (2020) Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: a multicentre study. *Eur J Radiol* 128:109041
- Xu L et al (2020) Low-dose chest X-ray image super-resolution using generative adversarial nets with spectral normalization. *Biomed Signal Process Control* 55:101600
- Yamashita R et al (2018) Convolutional neural networks: an overview and application in radiology. *Insights Imaging* 9(4):611–629
- Yin Z et al (2017) A deep normalization and convolutional neural network for image smoke detection. *IEEE Access* 5:18429–18438
- Zeng G (2020) On the confusion matrix in credit scoring and its analytical properties. *Commun Stat Theory Methods* 49(9):2080–2093
- Zhang K et al (2017) Beyond a gaussian denoiser: residual learning of deep CNN for image denoising. *IEEE Trans Image Process* 26(7):3142–3155
- Zhijie Y et al (2020) Bactran: a hardware batch normalization implementation for CNN training engine. *IEEE Embed Syst Lett* 13:29

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.