A Performance Evaluation of Convolutional Neural Networks for Face Anti Spoofing

Chaitanya Nagpal Shiv Ram Dubey
Computer Vision Group,
Indian Institute of Information Technology, Sri City
Andhra Pradesh-517646, India
chaitanya.n14@iiits.in, srdubey@iiits.in

Abstract

In the current era, biometric based access control is becoming more popular due to its simplicity and ease to use by the users. It reduces the manual work of identity recognition and facilitates the automatic processing. Face is one of the most important biometric visual information that can be easily captured without user cooperation in uncontrolled environment. Precise detection of spoofed faces should be on the high priority to make face based identity recognition and access control robust against possible attacks. The recently evolved Convolutional Neural Network (CNN) based deep learning technique has been proved as one of the excellent method to deal with the visual information very effectively. The CNN learns the hierarchical features at intermediate layers automatically from the data. Several CNN based methods such as Inception and ResNet have shown outstanding performance for image classification problem. This paper does a performance evaluation of CNNs for face anti-spoofing. The Inception and ResNet CNN architectures are used in this study. The results are computed over benchmark MSU Mobile Face Spoofing Database. The experiments are done by considering the different aspects such as depth of the model, random weight initialization vs weight transfer, fine tuning vs training from scratch and different learning rate. The favorable results are obtained using these CNN architectures for face anti-spoofing in different settings.

1. Introduction

In recent years, we have witnessed the growth and development of new and innovative methods for automatic authentication [32]. With the growth of data and the increasing awareness about the sensitivity of personal information, people have started to treat their privacy more seriously. The development of more robust and user-friendly authentication and access control devices is on high priority by utilizing the visual information such as fingerprint, facial, iris, and many more as compared to the password and token based devices [23]. The major challenge in any automatic access control method is the protection against malicious attacks by intruders [34]. Specifically, in face based authentication, the major challenges are to deal with the following three attacks, (i) printed photos, (ii) replay videos, and (iii) 3D videos. Face anti-spoofing is the field of study that tackles the above mentioned mentioned challenges in a robust and efficient manner.

The face anti-spoofing is one of the fundamental problem of biometric and computer vision. In initial years, the hand-designed feature based approaches were more common and utilized the characteristics like texture-based features, motion-based features and depth-based features [5]. The texture-based analysis exploited the fact that real face contains different texture and illumination pattern as compared to a plastic or LCD surface used to accomplish the attack. Maatta et al. [20] used a multi-scale local binary pattern (LBP) followed by a non-linear SVM to deal with such attacks. Chingovska et al. [3] also used the similar approach for the same problem. They extracted the LBP descriptors from a greyscale image and applied 3 classifiers on top of the LBP features to perform the classification. These methods are not efficient and require a lot of data pre-processing to be done. The motion based face anti-spoofing is also investigated by several researches by exploiting the fact that the most of face attacks happen with the use of stills and thus, lack the basic motion that can be used to differentiate a live subject from an image. Anjos et al. [1] utilized the motion relation between foreground and background to differentiate between a live face and an attacked face. Pereira et al. [4] used the LBP-TOP features containing space and time descriptors to encode the motion information along with the face texture. Kollreinder et al. [16] extracted the facial parts (e.g., left and right eyes, nose, left and right ears) by simplified optical flow and then modeled the liveliness of these parts through a short sequence of images. The noise in the face image is also treated as the important characteristics for face anti-spoofing with the fact that the noise level in attacked face is more due to the reconstruction process of any spoofing method. Zhang et al. [37] utilized the multiple Difference of Gaussian (DoG) filters to remove the noise and low-frequency information. They used the high frequency information to generate the feature vector for SVM classifier to distinguish between genuine and fake faces. Wen et al. [35] considered the 4 types of surface deformations such as specular reflection, blurriness features, chromatic moment and color diversity to generate the feature vector and used SVM classifier to classify the feature vector into real vs spoofed. The above discussed methods had several drawbacks like the need to utilize hand designed features and the limited performance of these methods.

Recent trends in computer vision have shown a gradual shift towards Convolutional Neural Networks (CNN) due to its characteristics like automatic learning and higher accuracy [13]. The CNN based approaches have been proven to be a very effective approach for different problems of visual information processing like object detection, semantic segmentation, image classification, biomedical analysis, image captioning, image coloring, biometric authentication, and many more [9]. In many scenarios, the performance of these methods even surpasses the human/expert level performance. ImageNet Large Scale Visual Recognition Challenge [27] has fostered the development of new and better CNN architectures over the years. The winning architectures like AlexNet [17] in 2012, VGGNet [28] and GoogleNet [29] in 2014, and ResNet [11] in 2015 brought a number of improvements and innovations to the field of object recognition. The task of object detection has also witnessed a series of improvements over the last few years through the evolution of CNN architectures like R-CNN [7], Fast R-CNN [6] and Faster R-CNN [26]. These approaches have made the object detection task not only faster than traditional methods but also improved the performance very drastically. The CNN architectures like Fully Convolutional Networks [19] and Mask R-CNN [10] have made the image segmentation much easier, intuitive and semantic. These approaches have gained very high improvement over its ancestral and hand-designed methodologies. The biomedical image processing area has also observed the immense improvement by using the CNN based methods in the problems like Colon Cancer Detection [14] and Radiologist-Level Pneumonia Detection [25], etc.

Some researchers have also explored CNNs for the biometric authentication and verification over the years as an alternative to traditional methods. Different CNN architectures are proposed for different biometric traits such as fingerprint, face, iris, etc. [21], [22], [31], [36], [24], [12]. Facial authentication systems cover a number of problems as discussed earlier and various attempts have been made recently to solve these problems. Recently, CNN is also being applied for face anti-spoofing and liveliness detection. Gragnaniello et al. [8] utilized the domain-specific knowledge to deal with robustness problem in CNN architecture for biometric spoofing detection. Li et al. [18] fine tuned the CNN over face spoofing datasets and then extracted the features and applied the principle component analysis (PCA) to reduce the dimensionality and finally the SVM is employed to do the classification into real vs spoofed face. Atoum et al. [2] utilized an ensemble of patch-based and depth-based CNN to perform the classification as well as liveliness detection in facial unlocking systems. All these methodologies proved that the CNNs can be used very effectively for the biometric anti-spoofing by automatically extracting the biometric features from training data.

Motivated by the success of CNNs in many visual information processing tasks in biometric and computer vision, this paper presents a performance evaluation of state-of-the-art CNN architectures such as Inception-v3, ResNet50 and ResNet152 for face anti-spoofing. The experiments are conducted to cover the various aspects of using the CNN for face anti-spoofing such as the depth of architecture, fine tuning and training from scratch, pre-trained weight transfer and random weight initialization, and different learning rates. This paper provides the best practices to utilize the CNN based approaches such as Inception-v3, ResNet50 and ResNet152 for face anti-spoofing problem.

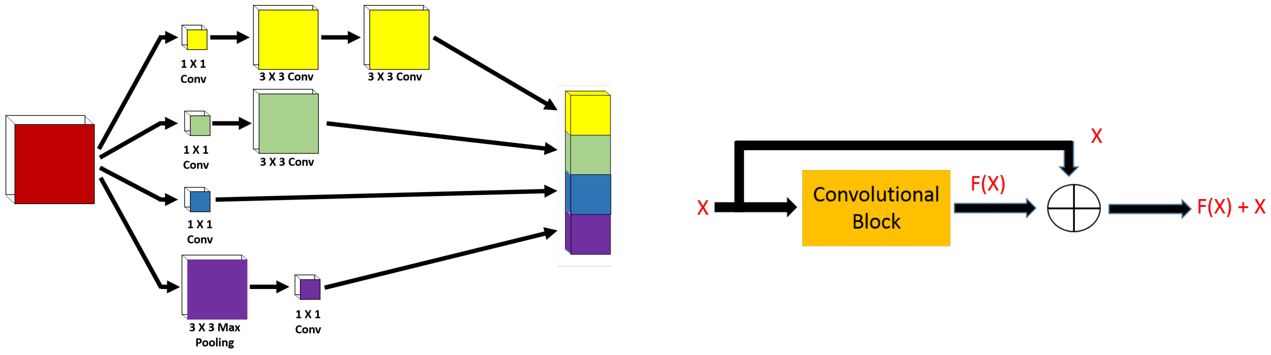
The rest of the paper is divided into various sections. Section 2 discusses about the state-of-the-art CNN architectures compared in this study. Section 3 describes the experimental setup including the framework of face anti-spoofing using CNN, hyperparameter settings, the database characteristics and data preprocessing performed. Section 4 presents the experimental results with detailed analysis from different perspective. Sections 4 concludes the paper with constructive suggestions for future initiatives.

2. CNN Architectures Used

As discussed in the earlier section, the Convolutional Neural Networks (CNNs) are the new trends in computer vision. The CNNs have shown immense improvements in image and video based classification problems. In this study, we conduct a performance evaluation of state-of-the-art CNNs such as Inception-v3, ResNet50 and ResNet152 for face anti-spoofing. This section provides an overview of Inception and ResNet modules.

2.1. Inception-v3 Module

In 2014, Szegedy et al. of Google Inc. [29] proposed GoogLeNet which won the ImageNet Large-Scale Visual



(a) Inception-v3 Module (b) Residual Module

Figure 1. The structure of Inception-v3 [30] and Residual [11] modules. These modules are stacked to form the deep network of Inception-v3 [30] and ResNet [11], respectively.

Recognition Challenge 2014 (ILSVRC14) [27] for classification and detection. The GoogLeNet is based on the inception module which basically combines the convolution outputs of varying filter sizes including 1×1, 3×3 and 5×5 with max pooling output. The original Inception module also uses the 1×1 bottleneck to reduce the complexity. Several inception modules are stacked over each other in GoogLeNet [29] to form a 22 layers deep network. The Inception module makes the GoogLeNet faster and efficient as compared to previous models like VggNet [28], etc. Just after 1 year (i.e., in 2015), Szegedy et al. proposed an Inception-v3 module [30] which is basically the redesign version of the original Inception module [29]. The Inception-v3 module increases the computational efficiency drastically as compared to the original Inception module by factorization of the bigger convolutions into smaller convolutions. In Inception-v3 module, each 5×5 convolution is replaced by two 3×3 convolutions which reduces the number of operations while covering the same receptive field. The Inception-v3 module is shown in Fig. 1(a). It computes four output volumes by applying (1) 1×1 convolution, (2) 1×1 convolution followed by 3×3 convolution, (3) 1×1 convolution followed by two 3×3 convolutions with different weights, and (4) 3×3 max pooling followed by 1×1 convolution, respectively for any input volume. Finally, all four output volumes are concatenated to form a single output volume. The dimension of output volume is same as the input volume by using the 1×1 bottleneck as depicted in Fig. 1(a).

2.2. Residual Module

During the evolution of CNN architectures over the years from AlexNet (8 layers) [17] in 2012 to VggNet (16 or 19 layer) [28] and GoogLeNet (22 layers) [29] in 2014, it is observed that the deeper networks perform better due to the increased complexity. Following the same line, He et al. of Microsoft research [11] conducted the experiment with 56-layer plain convolution architecture and found that the performance of 56-layer is worse than 20-layer. They analyzed that the deeper network is very hard to optimize and leads to the decreased performance. In order to overcome this optimization issue, they proposed to learn the residual instead of the plain transformation. They introduced the ResNet [11] architecture which uses the residual block to pass more information towards the last layers. The residual unit basically facilitates to provide the crucial information to next unit which is actually lost in convolution step. The structure of residual unit is shown in Fig. 1(b), here the convolutional block represent two convolution operation, X is the input volume to residual unit, $F(X)$ is the output volume of convolutional block, and $F(X) + X$ is the output volume of residual block/unit. It can be perceived from Fig. 1(b) that the residual unit learns $F(X)$ which is basically the residual of output from input. Several residual blocks are stacked in ResNet for deeper architecture. Based on the number of residual blocks, the depth of ResNet is different. The ResNet architecture was also the winner of ImageNet Large-Scale Visual Recognition Challenge 2015 (ILSVRC15) [27] classification task and achieved 3.57% error which surpasses the human level performance. The ResNet also secured the 1^{st} positions for the ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation tasks. In this study, ResNet50 and ResNet152 architectures with 50 and 152 layers, respectively are used for performance evaluation over face anti-spoofing task.

3. Experimental Setup

This section describes the performance evaluation experimental setup in terms of the face anti-spoofing framework using CNN, hyperparameter settings, evaluation criteria and face spoofing database used.

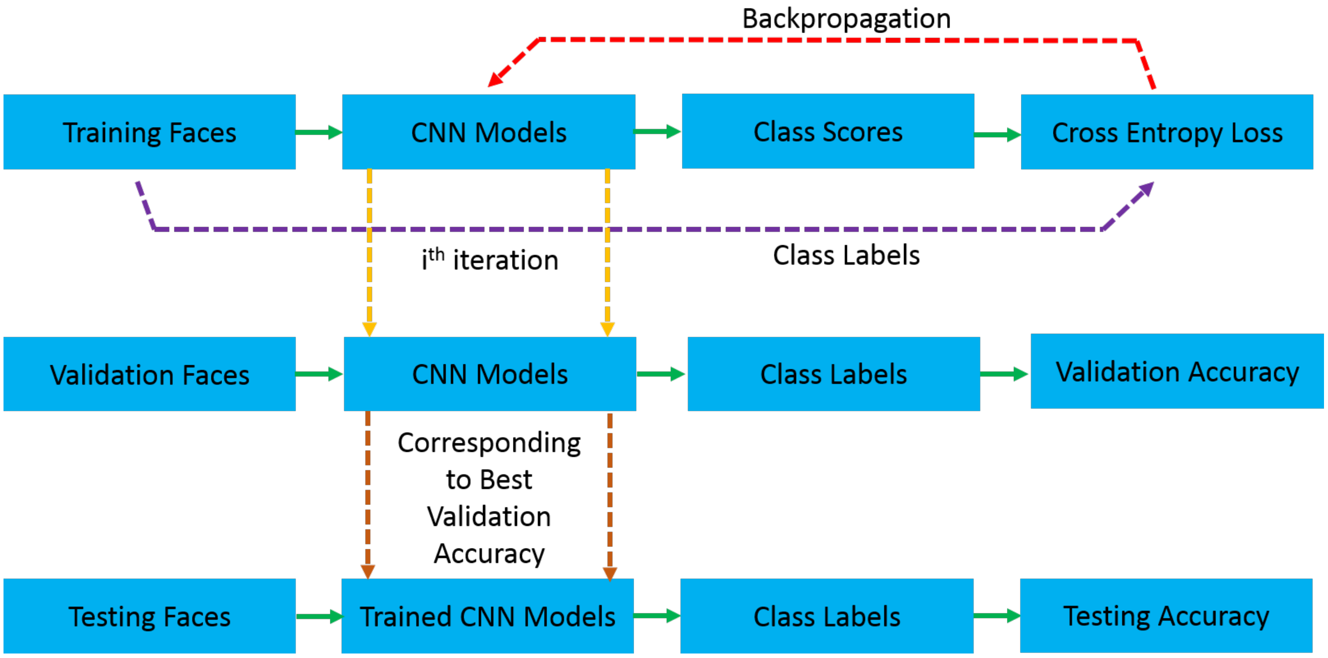


Figure 2. Training, validation and testing framework for face anti-spoofing using CNN models. The Inception-v3, ResNet50 and ResNet152 models are used in this paper.

3.1. CNN based Face Anti-spoofing

The face anti-spoofing is considered as the two-class classification problem in this paper. The two classes are real face class and spoofed face class. Fig. 2 shows the training and testing framework for real and spoofed face classification using CNN model such as Inception-v3, ResNet50 and ResNet152. During training phase, the CNN model predicts the class score for training images, computes the categorical cross-entropy loss, and finally update the weights of network using gradient descent method by back-propagating the gradient w.r.t. loss function. In every epoch, the learned weights using training images are used to generate the class scores and classification accuracy over validation images. Once training is done, the learned weights corresponding to highest validation accuracy is used for testing. During the testing phase, the trained CNN model generates the class scores for input face image and predicts the class corresponding to the highest class score.

The experiments are performed over a desktop computer system having an Intel Core i7-7700 CPU, 32 GB RAM (i.e., 2 × 16 GB RAM) and one 8 GB NVIDIA Zotac GeForce GTX 1080 GPU. The programs are written using the Keras open source neural network library in Python running on top of the TensorFlow deep learning framework. The Adam optimizer [15] has been proved to be a suitable by-default stochastic optimization technique in most of the problems of neural network. Thus, in our experiments also, the Adam optimizer is used to train all CNN models. The experiments are conducted with following different setups, (1) the weights are transferred from the pre-trained weights computed over ImageNet database, (2) the weights are initialized randomly, (3) only fully connected layer is trained and weights of other layers are freezed, (4) all layers are trained irrespective of the initialization, and (5) two learning rates (i.e., 10^{-3} and 10^{-5}) are used without any learning rate annealing. In order to evaluate the performance of different models for different hyperparameter settings, the training, validation and testing accuracies are computed. The convergence time is also computed in terms of the minimum number of epochs needed to get the highest result.

3.2. Database Used

For the course of the performance evaluation in this paper, we used the benchmark MSU Mobile Face Spoofing Database (MFSD) [35]. It consists of 8 videos of 35 subjects. The video sets for each user consist of 2 real videos and 6 fake videos captured through various devices. For our experiments, first the videos are converted into frames, then the face in frames is localized by using the Viola Jones Harr Cascade [33], and finally the extracted faces are randomly split into three sets including training, validation and testing. This procedure is depicted in Fig. 3. The training set

Table 1. The statistics of MFSD database [35] in terms of the number of samples in training, validation and testing sets along with the proportion of samples for different attacks.

	Fake							Real		
Device	Phone	Laptop	Phone	Laptop	Phone	Laptop	Total	Phone	Laptop	Total
Attack	Printed	Printed	Phone	Phone	Tablet	Tablet		NA	NA	
Testing	410	376	507	516	546	516	2871	414	387	801
Validation	411	350	469	486	447	485	2648	426	414	840
Training	7201	6688	8862	9026	9461	9498	50736	7733	7812	15545
+Flipped	7201	6688	8862	9026	9461	9498	50736	7733	7812	15545
= Total	14402	13376	17724	18052	18922	18996	101472	15466	15624	31090

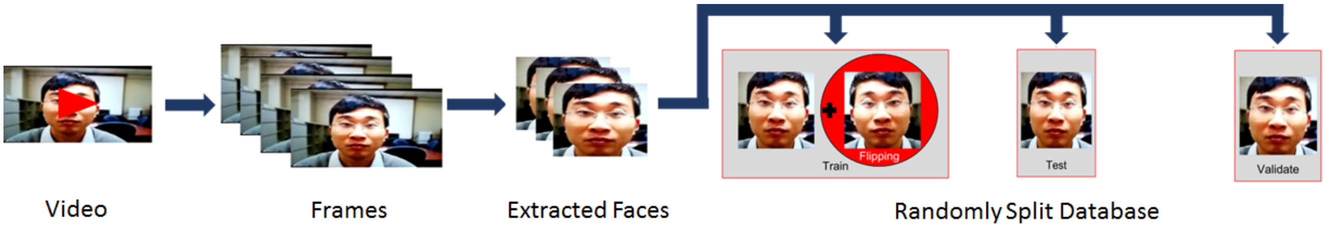


Figure 3. MSU Mobile Face Spoofing Database (MFSD) [35] preparation including frames extraction from video, face localization in frames using Viola Jones Harr Cascade [33], and sample split into training, validation and testing sets for the experiments in this study. Note that, only horizontal flipping is applied over training images for data augmentation.

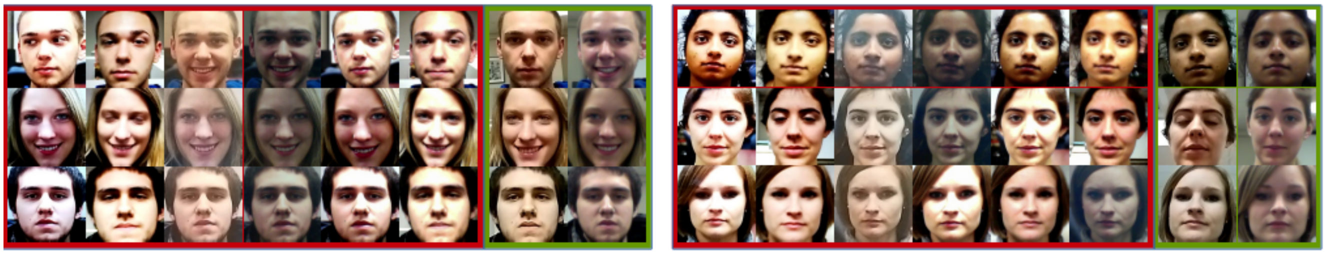


Figure 4. The sample faces after cropping from MFSD database [35]. The faces in a row are corresponding to a particular subject. The images inside the Red and Green rectangular boxes contain the spoofed and real faces, respectively.

is flipped horizontally to apply the data augmentation. The sample faces extracted from MFSD database are displayed in Fig. 4 before applying flipping. Each row corresponds to the faces of a particular subject. The spoofed and real faces are enclosed within the Red and Green rectangular boxes, respectively.

The extracted face images are used to train, validate and test the Inception-v3, ResNet50 and ResNet152 models. A total of 73441 images are extracted from the videos of all subjects. These images are further distributed randomly into training, testing and validation sets including 66281, 3672 and 3488 images, respectively. The images in the training set are augmented by horizontal flipping, thus doubling the training dataset size to 132562. A complete statistics of the used MFSD database is presented in Table 1 including the number of images of different attacks.

4. Performance Evaluation and Observations

In order to find the best practices for face anti-spoofing using CNN architectures such as Inception-v3, ResNet50 and ResNet152, we performed several experiments and the analyzed the results. We compared different CNN models in this section in terms of the accuracies, rate of convergence and other factors such as weight transfer, random weight initialization, fine tuning, training from scratch and different learning rate.

4.1. Test Accuracy Comparison

The different models trained on the same database with varying parameters have shown drastic variances in performance. The Fig. 5 shows the comparison among the accuracy of the Inception-v3, ResNet50 and ResNet152 models

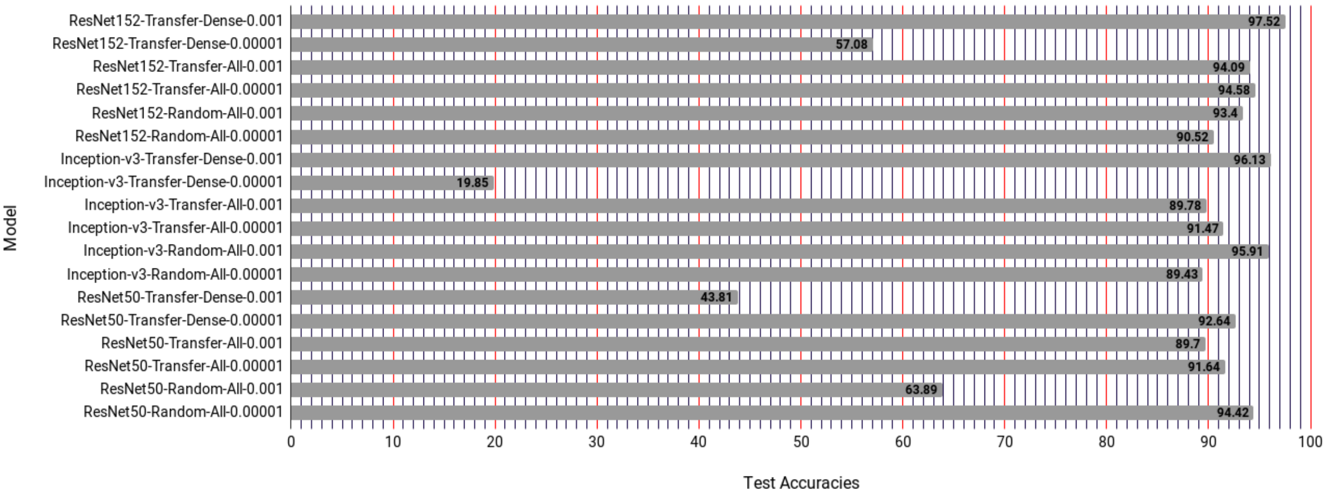


Figure 5. Test accuracy corresponding to the trained weights of highest validation accuracy for different models explored in this study in different settings. Following is model name convention: ModelName-WeightInitializationType-TrainableLayers-LearningRate. Here, ‘Transfer’ refers to the weight initialization by transferring from pre-trained ImageNet weights of that model, ‘Random’ refers to random weight initialization, ‘Dense’ corresponds to the training of dense layers only, and ‘All’ corresponds to the training of all layers.

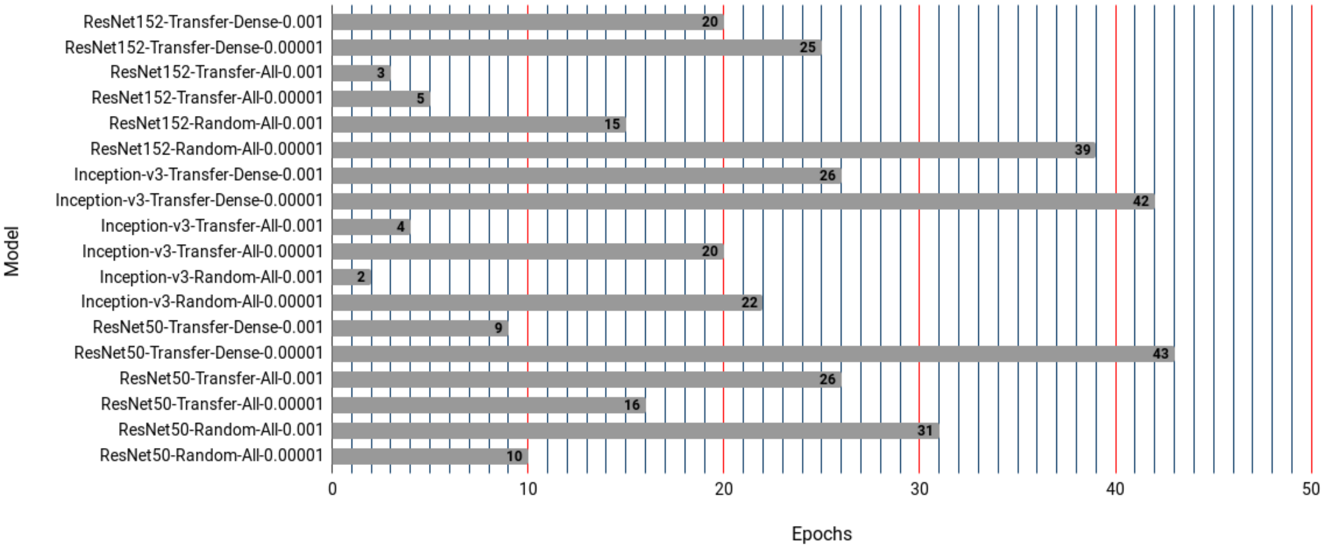


Figure 6. The minimum number of epochs taken to reach the maximum validation accuracy. The naming convention is similar to Fig. 5.

obtained over the test set corresponding to the highest validation accuracy. Highest test accuracy observed over the MFSD database is 97.52% for the ResNet152 model trained through fine tuning of dense layers using ImageNet challenge ResNet152 weights at a learning rate of 10^{-3}. For the same ResNet152, the test accuracy decreases on decreasing the learning rate in case of fine tuning of dense layers. However, the performance of ResNet152 increases after decreasing the learning rate when all the layers are trained. Comparing the weight initialization methods for ResNet152, it is evident that the test accuracy increases for weight transfer as compared to the random weight initialization while keeping all the other settings same. On the other hand, for Inception-v3 model, the results slightly vary. The highest test accuracy noted for Inception-v3 is 96.13% which is achieved when the model is fine tuned on Imagenet challenge weights at a learning rate of 10^{-3}. Even in the case of Inception-v3, it can be observed that by decreasing the learning rate for the same parameters causes an increase in the accuracy in general. However, when the Inception-v3 is fully trained with Imagenet weights, a higher accuracy is achieved at lower learning rate (i.e., 10^{-5}). Comparing the weight initialization methods for Inception-v3, we observed that the random initialization works better for

Table 2. The training, validation and testing performance comparison among Inception-v3, ResNet50 and ResNet152 models in terms of the accuracy, convergence rate, and varying parameters like initial weights, number of trainable layers and learning rate. In this table, the 'Epochs' is the number of epochs for highest validation accuracy.

Base Model	Initial Weights	Trainable Layers	Learning Rate	Training Accuracy	Validation Accuracy	Testing Accuracy	Epochs
Resnet 152	Imagenet	Dense	0.001	92.63	99.59	97.52	20
Resnet 152	Imagenet	Dense	0.00001	94.06	55.96	57.08	25
Resnet 152	Imagenet	All	0.001	90.58	96.24	94.09	3
Resnet 152	Imagenet	All	0.00001	93.44	98.54	94.58	5
Resnet 152	Random	All	0.001	90.86	97.94	93.40	15
Resnet 152	Random	All	0.00001	93.38	94.18	90.52	39
Inception-v3	Imagenet	Dense	0.001	94.63	96.47	96.13	26
Inception-v3	Imagenet	Dense	0.00001	91.88	20.56	19.85	42
Inception-v3	Imagenet	All	0.001	90.97	99.23	89.78	4
Inception-v3	Imagenet	All	0.00001	93.60	95.63	91.47	20
Inception-v3	Random	All	0.001	91.33	97.07	95.91	2
Inception-v3	Random	All	0.00001	93.95	98.57	89.43	22
ResNet 50	Imagenet	Dense	0.001	91.34	72.88	43.81	9
Resnet 50	Imagenet	Dense	0.00001	92.66	94.26	92.64	43
ResNet 50	Imagenet	All	0.001	91.26	98.71	89.70	26
ResNet 50	Imagenet	All	0.00001	96.67	98.22	91.64	16
ResNet 50	Random	All	0.001	93.72	93.84		5
ResNet 50	Random	All	0.00001	92.80	97.88	94.42	10

higher learning rate while transfer learning works better for lower learning rate. ResNet50 achieves highest accuracy of 94.42% when trained from random weights at a learning rate of 10^{-5}. In general it can be observed that ResNet50 performs better when trained with a lower learning rate.

4.2. Convergence Rate Comparison

The training of different CNN models exhibit the varying rate of convergence as shown in the Fig. 6. It is affected by several factors like the model type, model complexity, model size, number of trainable layers, training method, etc. In general, the transfer learning is proved to be faster than random weight initialization based training for the same model. The Inception-v3 and ResNet50 models experience the gain in training time for transfer learning at lower learning rate of 10^{-5}. It is evident from the Fig. 6 that the ResNet152 model takes the most amount of time when initialized with random weights and trained with a learning rate of 10^{-5}. One important observation of ResNet50 model is that when trained at a learning rate of 10^{-5}, the model converges faster as compared to the learning rate of 10^{-3} while the training is dome through transfer learning with initial weights transferred from Imagenet challenge. On an average the Inception-v3, ResNet50 and ResNet152 models take about 24.1, 17.8, 18.17 epochs, respectively to converge.

4.3. Training, Validation and Testing Results Comparison

The training, validation and testing accuracy of Inception-v3, ResNet50 and ResNet152 models is summarized in Table 2. Comparing the validation accuracy, it can be observed that the highest validation accuracy registered is 99.59% for ResNet152 through transfer learning with a learning rate of 10^{-3}. The same setup also achieves the highest testing accuracy as discussed earlier. It is also observed that the least validation accuracy for both Inception-v3 and ResNet152 models is observed through transfer learning at a learning rate of 10^{-5}. Whereas, under the same conditions, the ResNet50 performs very well with a validation accuracy of 94.26%.

5. Conclusion

In this paper, a performance comparison is conducted for face anti-spoofing by using the CNN models. The recently discovered and state-of-the-art CNN architectures such as Inception-v3, ResNet50 and ResNet152 are used in this study. The experiments are performed over MSU Mobile Face Spoofing Database (MFSD). The MFSD database is partitioned into training, validation and testing sets. The results are computed against the epoch number corresponding to the highest validation accuracy achieved. The performance comparison is done w.r.t. different conditions such as depth of ResNet model, weight initialization methods, number of trainable layers and learning rate. The ResNet152 model is the best suited one for face anti-spoofing task when only dense layers are trained with weight initialization through ImageNet weight transfer and learning rate of 10^{-3}. It is also observed that the lower learning rate is better for ResNet152, whereas higher learning rate is better for ResNet50. The Inception-v3 gives an acceptable trade-off between accuracy and rate of convergence. It is also revealed from the results that the transfer learning over all layers leads to the faster rate of convergence for ResNet152 and Inception-v3 models, whereas the same setting is against the ResNet50 model. Based on the observations of this study of face anti-spoofing using CNN models, it is suggested to utilize the deeper models at lower learning rates with transfer learning for last fully connected layers. In case of limited computational resources, it is recommended to use Inception-v3 architecture with the similar setting of above mentioned ResNet152 such as transfer learning for dense layers at lower learning rate.

References

[1] A. Anjos and S. Marcel. Counter-measures to photo attacks in face recognition: a public database and a baseline. In *Biometrics (IJCB), 2011 international joint conference on*, pages 1–7. IEEE, 2011.

[2] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu. Face anti-spoofing using patch and depth-based cnns. In *Biometrics (IJCB), 2017 IEEE International Joint Conference on*, pages 319–328. IEEE, 2017.

[3] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *Biometrics Special Interest Group (BIOSIG), 2012 BIOSIG-Proceedings of the International Conference of the*, pages 1–7. IEEE, 2012.

[4] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel. Lbp- top based countermeasure against face spoofing attacks. In *Asian Conference on Computer Vision*, pages 121–132. Springer, 2012.

[5] J. Galbally, S. Marcel, and J. Fierrez. Biometric antispoofing methods: A survey in face recognition. *IEEE Access*, 2:1530–1552, 2014.

[6] R. Girshick. Fast r-cnn. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 1440–1448. IEEE, 2015.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation, 2013.

[8] D. Gragnaniello, C. Sansone, G. Poggi, and L. Verdoliva. Biometric spoofing detection by a domain-aware convolutional neural network. In *Signal-Image Technology & Internet-Based Systems (SITIS), 2016 12th International Conference on*, pages 193–198. IEEE, 2016.

[9] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 2017.

[10] K. He, G. Gkioxari, P. Dollr, and R. Girshick. Mask r-cnn, 2017.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] G. Jaswal, A. Nigam, and R. Nath. Deepknuckle: revealing the human identity. *Multimedia Tools and Applications*, 76(18):18955–18984, 2017.

[13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[14] P. Kainz, M. Pfeiffer, and M. Urschler. Semantic segmentation of colon glands with deep convolutional neural networks and total variation segmentation. *arXiv preprint arXiv:1511.06919*, 2015.

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] K. Kollreider, H. Fronthaler, and J. Bigun. Non-intrusive liveness detection by face images. *Image and Vision Computing*, 27(3):233–244, 2009.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[18] 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 *Image Processing Theory Tools and Applications (IPTA), 2016 6th International Conference on*, pages 1–6. IEEE, 2016.

[19] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[20] J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *Biometrics (IJCB), 2011 international joint conference on*, pages 1–7. IEEE, 2011.

[21] D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcão, and A. Rocha. Deep representations for iris, face, and fingerprint spoofing detection. *IEEE Transactions on Information Forensics and Security*, 10(4):864–879, 2015.

[22] R. F. Nogueira, R. de Alencar Lotufo, and R. C. Machado. Fingerprint liveness detection using convolutional neural networks. *IEEE transactions on information forensics and security*, 11(6):1206–1213, 2016.

[23] L. O'Gorman. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12):2021–2040, 2003.

[24] K. B. Raja, R. Raghavendra, V. K. Vemuri, and C. Busch. Smartphone based visible iris recognition using deep sparse filtering. *Pattern Recognition Letters*, 57:33–42, 2015.

[25] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, et al.

Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

[26] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[27] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[28] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions, 2014.

[30] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[31] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[32] J. Unar, W. C. Seng, and A. Abbasi. A review of biometric technology along with trends and prospects. *Pattern recognition*, 47(8):2673–2688, 2014.

[33] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.

[34] J. Wayman, A. Jain, D. Maltoni, and D. Maio. An introduction to biometric authentication systems. In *Biometric Systems*, pages 1–20. Springer, 2005.

[35] D. Wen, H. Han, and A. K. Jain. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, 10(4):746–761, 2015.

[36] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[37] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li. A face antispoofing database with diverse attacks. In *Biometrics (ICB), 2012 5th IAPR international conference on*, pages 26–31. IEEE, 2012.