

Alzheimer's Disease Classification Using Brain MRI Based on 2D CNN and Transfer Learning



Santosh Kumar Sahoo, Sukanta Das, and Sandeep Kumar Panda

Abstract In the present scenario, it is believed that around 44 M people are having Alzheimer's disease and the data is estimated to double in the year of 2030. However, there is little research that has been done in this field. In this regard, we design a model to recognize Alzheimer's disease instances, particularly in the beginning phase using Magnetic Brain Imaging (MRI) scans. For validation, the entire data set consists of MRI scans which have been collected from various repositories with each label verified. Alzheimer's Disease dataset is partitioned into 2 set i.e., training set and testing set and these sets are further divided into 4 classes i.e., Mild Demented, Moderate Demented, Non-Demented and Very Mild Demented. Our aim was to make a model which gives the highest accuracy in predicting the stage of Alzheimer's Disease (AD). AD Prediction is done using Deep learning (DL) models. The prediction process is done by uploading our dataset into Google Drive then our model takes the MRI scans as input and detects the state of Alzheimer's Disease. We have used Deep Learning models and analyze whose accuracy is highest. We used a 2D Convolutional Neural Network which takes the MRI images as input. It computes and returns the stage of Alzheimer's Disease. For more accuracy, we have used one Machine Learning model i.e. Support Vector Machine (SVM) with have an accuracy of 50.68, CNN model with 68.33% accuracy and various Transfer Learning models like Visual Geometry Groups(VGG)16, VGG19, Resnet50 and InceptionV3 with 62.7%, 63.25%, 68.49% and 56.13% respectively from Keras API. Our results show that Resnet50 gave the highest accuracy comparison to others model.

S. K. Sahoo · S. Das

Department of Computer Science and Engineering, Faculty of Science and Technology (IcfaiTech), ICFAI Foundation for Higher Education (Deemed to be University), Hyderabad, Telangana, India

S. K. Panda (✉)

Department of Data Science and Artificial Intelligence, Faculty of Science and Technology (IcfaiTech), ICFAI Foundation for Higher Education (Deemed to be University), Hyderabad, Telangana, India

e-mail: skpanda00007@gmail.com

Keywords Alzheimer’s Disease · Stages of Alzheimer’s Disease · 2D convolutional neural network · Transfer Learning · VGG16 · VGG19 · Resnet50 · SVM · Keras API

1 Introduction

Alzheimer’s disease is a neurological condition that causes the death of brain cells and the brain gets shrink. It is progressive in nature. A continual deterioration in mental, behavioral, and social abilities that undermines a person’s capacity to operate independently- is the most common cause of dementia [1]. Mostly people who are above 65 are more prone to have this disease (as shown in Fig. 1) and very less number of people in the age between 30–65 have this disease which is known as early-onset. In the final stage of disease, individuals lose their ability to react to their surroundings, converse, and eventually regulate their mobility. They may still pronounce words and expressions, but it becomes more difficult to communicate. Research on Alzheimer’s disease has been intensified due to the increasing number of patients per year and to remove manual errors that are caused by the human eye.

In past decade, we have seen the development of Deep Learning techniques in medical related fields. When we compare Deep Learning and convolutional neural networks (CNNs) to traditional ML algorithms, they have dramatically improved performance in this field.

Basically, CNN is a Deep Learning algorithm that takes in image as input (as shown in Fig. 1) and learn from it by assigning learnable biases and weights to various objects of the image. It has one input layer at the starting which consists of many neurons where each neuron represents the pixel value of the input image and the last layer of the network contains the labels which are present in the form of One-hot Encoding. The layers in between the input layer and output layer are called Hidden Layers. Each neuron in the hidden layer takes input from the patch. Then it computes weighted sums and applies biases. The above definition can be represented by this equation,

$$\text{Weighted sum output for a patch} = \text{act}\left(\sum_{i=1}^n \sum_{j=1}^n W_{ij} x_{i+p,j+q} + b\right)$$

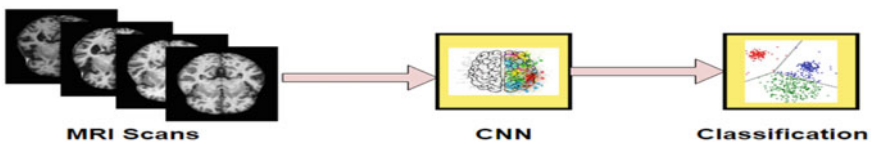


Fig. 1 CNN Representation

where n is size of the filter act is an activation function W_{ij} is a weight matrix, b is bias p and q represents the (p,q) neuron in the hidden layer. There are three main components that are core to a CNN. First part is convolutional operation, it allows us to generate feature maps or detect features in our image. The second part is non-linearity, it helps us to deal with the features which are highly non-linear. And lastly, we have pooling or down sampling which allows us to scale down the size of each feature map.

CNNs have covered some extraordinary ground in image classification which plays an important role in the area of medical field. So, we have proposed various Transfer Learning techniques and CNNs for Detection of stages in Alzheimer's disease. Basically, it will mark the size of the affected part in the brain and according to that the model will draw a conclusion and place that image in 1 out of 4 stages.

2 Related Work

From the past two decade, there have been numerous published methods in the area of machine learning, especially in DL since 2013. DL models have grown prominently in medical image processing since 2013, when the research on new architectures in neural network gained momentum [1]. Deep learning technologies have shown groundbreaking performance in various fields in recent years (like visual object recognition, human action recognition, object tracking, image restoration, denoising, segmentation tasks, audio classification, brain-computer interaction etc.) [2]. Following the achievements of DL in models categorizing two dimensional photos, an increasing percentage of studies had attempted to integrate DL with medical images [3, 4]. DL architectures, such as convolutional neural network, may interpret hidden information in neuroimaging data, find linkages between different sections of an image, produce total cognition, and efficiently capture disease-related pathologies [1]. Deep learning methods have been effectively applied to sMRI, fMRI, PET, and DTI. The most prevalent method of identifying Alzheimer's disease, based on our research, is MRI, which motivated us for using MRI scans in this paper. Despite the fact that many researchers have used deep neural network to train their model from the start, deep models are frequently impossible to apply because of prolong convergence and huge dataset is necessary [5]. Commonly, we apply pre-trained CNNs on a single problem specific task as initialization step, and then we train the model again for new job by fine-tuning their last layers [6, 7]. The concept of "transfer learning" has proven to be an effective way to train a huge architecture without overfitting it. It has shown to be faster and more effective than traditional training, especially for cross-domain tasks [8–10]. In the ImageNet Database, Murala et al. [11] classified 1000 image classes. Different researchers later suggested numerous variations and advancements of CNN for Object identification and Image categorization for example VGG [12–16], ResNet [17–19], GoogLeNet [23–25], R-CNN [26–28], InceptionV3 [20–22] etc. Although with quicker training, its likely that

spatial correlations between slices are prone to loss. Transfer learning models can match the accuracy of a three-dimensional CNN model trained from scratch.

3 Material Methods

3.1 Cnn

As we all know, CNN is really effective when it comes to image classification because multiple convolutional filters are operating in all layers of a CNN, scanning the entire feature matrix and performing dimensionality reduction. Figure 2 shows the architecture of our proposed CNN model. In this model we have used a 3-layer Convolutional network with max pooling layer stacked between two convolutional layers. After that we have flattened the input and passed the resultant to a fully connected layer. I have also used Dropout layer and Batch normalization for Regularization. And finally, at the end I have used a dense layer with Softmax activation function as my output layer where,

$$\text{Softmax}(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

After that, we compiled the program using cross entropy loss function and Adam optimizer.

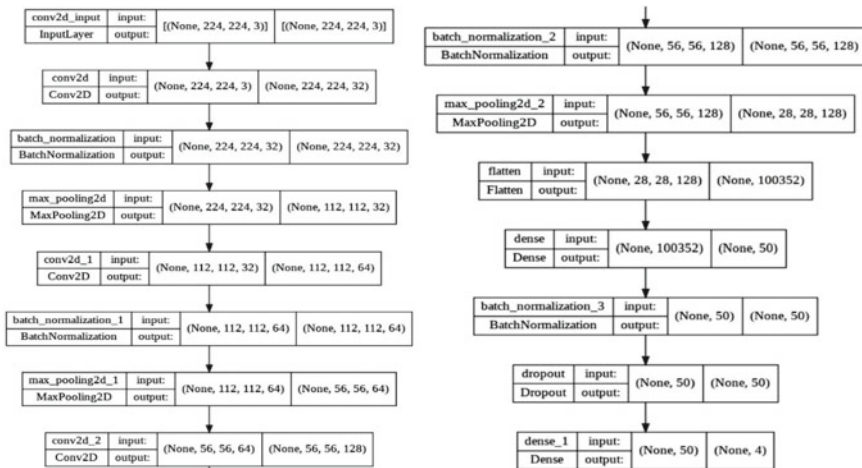


Fig. 2 Architecture of our CNN Model

3.2 *Resnet50*

A residual neural network50, also known as Resnet50. It is very efficient as it avoids the vanishing gradient problem by developing deeper networks than other plain networks while also determining the optimal number of layers.

We have used a pre-trained ResNet50 model because the weights and biases are already trained in millions of images. Then we iterate through all the layers and then each layer is set to non-trainable. By doing this we freeze the trainable parameters (weights and biases) in all the layers of the model so that they are not retrained whenever we go through the training process. We have also added two fully connected layers in the last for classification using Softmax function. At last, we have compiled the model using cross entropy loss function and Adam optimizer.

3.3 *InceptionV3*

InceptionV3 is a convolutional neural network architecture which is vividly used for image classification. It has high efficiency, computationally less expensive and uses auxiliary classifiers as regularizes. The basic idea behind inception was rather than choosing the kernel size and no. of filters and pooling etc., we basically use all of them together and stack the outputs of all of these together and the algorithm will learn which combinations to choose. The problem with such a model is that there are too many computations need to be done even in just one block. So to solve this issue, we make use of 1×1 convolutional. 1×1 convolutions reduce the number of computations significantly.

We have used a pre-trained InceptionV3 model because the weights and biases are already trained in millions of images. Then we iterate through all the layers and then each layer is set to non-trainable. By doing this we freeze the trainable parameters (weights and biases) in all the layers of the model so that they are not retrained whenever we go through the training process. We have also added two fully connected layers in the last for classification using Softmax function. At last, we have compiled the model using cross entropy loss function and Adam optimizer.

3.4 *Vgg16*

Visual Geometry Group16, also known as VGG16 is a cutting-edge transfer learning model. It has sixteen layers. It was constructed as a deep convolutional neural network model, outperforms baselines on a wide range of datasets outside of ImageNet VGG16 is profoundly used image recognition model nowadays. Figure 3 represents the architecture of our proposed VGG16 model. We have used a pre-trained VGG16 model as saves the time required for computation of the weights and biases.

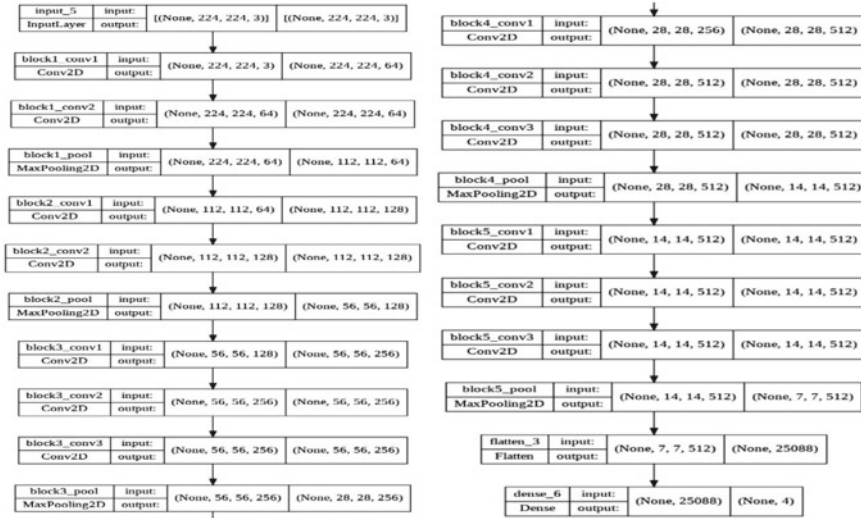


Fig. 3 Architecture of proposed VGG16 model

We iterate through all the layers and then each layer is set to non-trainable. By doing this we freeze the trainable parameters (weights and biases) in all the layers of the model so that they are not retrained whenever we go through the training process. We have also added another dense layer for classification using Softmax function as shown in Fig. 3. At last, we have compiled the model using cross entropy loss function and Adam optimizer.

3.5 Vgg19

Visual Geometry Group19, also known as VGG19 is the advanced version of VGG16. The motivation behind using this model is because of its high accuracy and faster training speed. VGG19, which was constructed as a deep convolutional neural network model, outperforms baselines on a wide range of datasets outside of ImageNet. VGG19 is one of the most widely used image-recognition models nowadays.

Figure 4 represents the architecture of our proposed VGG19 model. We have used a pre-trained VGG19 model as saves the time required for computation of the weights and biases. Then we iterate through all the layers and then each layer is set to non-trainable. By doing this we freeze the trainable parameters (weights and biases) in all the layers of the model so that they are not retrained whenever we go through the training process. We have also added two dense layer for classification using Softmax function as shown in the Fig. 3. At last, we have compiled the model using cross entropy loss function and Adam optimizer.

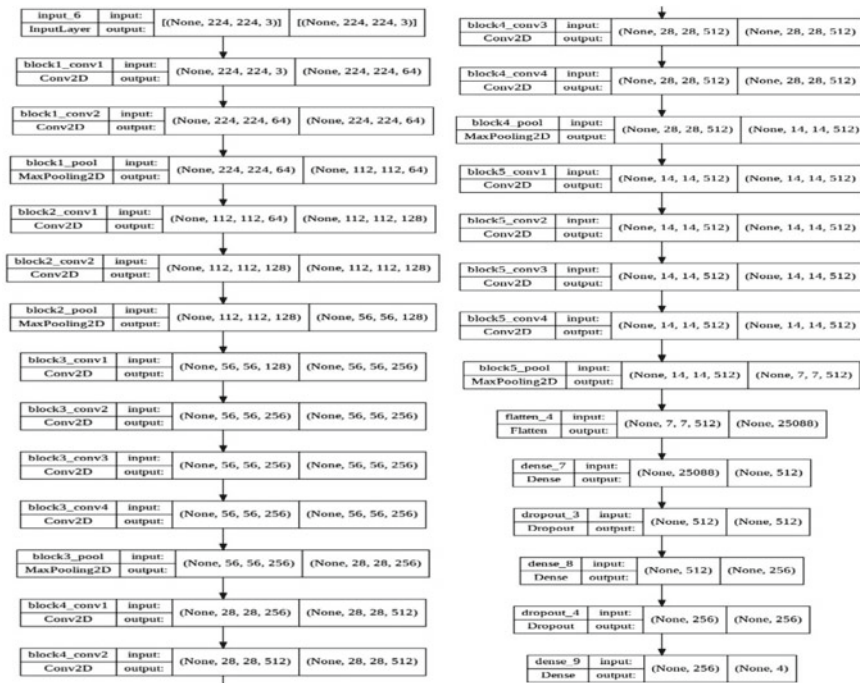


Fig. 4 Architecture of our proposed VGG19 model

4 Experiments and Results

4.1 Dataset

In both the training and test sets, the MRI dataset has a collection of 4 different types of scans. A total of 6400 images classified into 4 groups Mild Demented, Very Mild Demented, Non-Demented, Moderate Demented.

The images in the dataset were pre-processed to be 224 (W) * 224 (H) * 3 pixels in size (color channel).

4.2 Analysis of the Output

The training set is used to determine the accuracy and loss after each epoch, and it has 100 steps/epoch with a batch size of 64 out of 6400 images for Alzheimer’s disease.

For the results of our dataset, F1-score is a performance metric that we calculated.

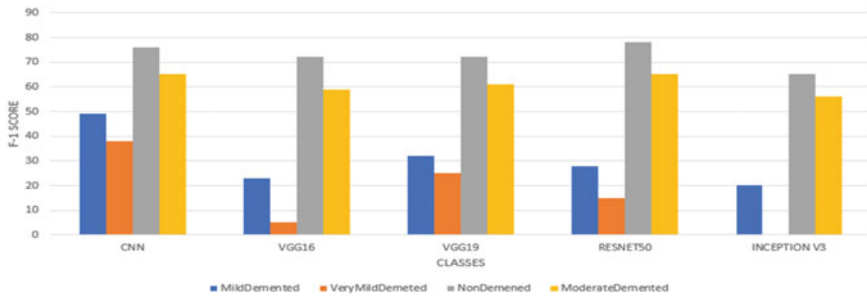


Fig. 5 F-1 score of all proposed models

$$F_1score = \frac{2TP}{2TP + FP + FN}$$

The F1 score is defined as, “harmonic mean of accuracy and sensitivity”. The most favorable value is 1, which is the aim of this research. The goal of this research is to get the most favorable value i.e., 1.

A true positive (TP) is the proportion of correct predictions by the model. For an instance, when the model predicts the class of Alzheimer’s disease correctly. Similarly, when the model correctly predicts a particular case of an Alzheimer’s disease not properly afflicted by it, then is called True negative (TN). When the model inaccurately predicts a particular class of an AD as an analogous class of AD, this is known as false positive (FP). FP is also known as TYPE-1 error. And lastly, when a model inaccurately predicts a particular case of an AD class to be analogous to another class is called false negative (FN). FN is also known TYPE-2 error.

The following are the results we got after applying different models (Fig. 5).

4.3 Output of Our Model

We put our model for testing with an unseen dataset i.e., test dataset. Total MRI scans in the testing data were 1279. Table 1 shows the accuracy for test dataset using various models:

Table 1 Testing Accuracy of our model

Model	Testing accuracy
SVM	50.68
CNN	68.33
VGG16	62.70
VGG19	63.25
RESNET	68.49
INCEPTION V3	56.13

5 Conclusion

In this paper, we used various Transfer Learning models to correctly predict the cases of AD. On the same dataset, different CNNs were trained. Our proposed model gave an accuracy of 68.49% on testing dataset using ResNet50 architecture. Performance can be further increase by applying Data Augmentation, Hyper parameter tuning and CNN architecture is well modified.

References

1. V. Wegmayr and D. Haziza, "Alzheimer Classification with MR images: Exploration of CNN Performance Factors," in Proceedings of the 1st Conference on Medical Imaging with Deep Learning (MIDL), 2018, pp. 1–7.
2. Sanghi, P., Panda, S.K., Pati, C., Gantayat, P.K. (2022). Learning Deep Features and Classification for Fresh or off Vegetables to Prevent Food Wastage Using Machine Learning Algorithms. In: Satapathy, S.C., Peer, P., Tang, J., Bhateja, V., Ghosh, A. (eds) Intelligent Data Engineering and Analytics. Smart Innovation, Systems and Technologies, vol 266. Springer, Singapore. https://doi.org/10.1007/978-981-16-6624-7_44
3. Panda SK, Sathya AR, Mishra M, Satpathy S (2019) A supervised learning algorithm to forecast weather conditions for playing cricket. International Journal of Innovative Technology and Exploring Engineering 9(1):1560–1565
4. Bhalerao, V., Panda, S.K., Jena, A.K. (2021). Optimization of Loss Function on Human Faces Using Generative Adversarial Networks. In: Bandyopadhyay, M., Rout, M., Chandra Satapathy, S. (eds) Machine Learning Approaches for Urban Computing. Studies in Computational Intelligence, vol 968. Springer, Singapore. https://doi.org/10.1007/978-981-16-0935-0_9
5. Panda SK, Swain SK, Mall R (2015) An investigation into usability aspects of E-Commerce websites using users' preferences. Advances in Computer Science: An International Journal 4(1):65–73
6. Panda, S.K., Swain, S.K., Mall, R. (2015). Measuring Web Site Usability Quality Complexity Metrics for Navigability. In: Jain, L., Patnaik, S., Ichalkaranje, N. (eds) Intelligent Computing, Communication and Devices. Advances in Intelligent Systems and Computing, vol 308. Springer, New Delhi. https://doi.org/10.1007/978-81-322-2012-1_41
7. Panda SK (2014) A Usability Evaluation Framework for B2C E-Commerce Websites. Comput. Eng. Intell. Syst. 5(3):66–85
8. Sandeep Kumar Panda, Vaibhav Mishra, R. Balamurali, Ahmed A. Elngar, Artificial Intelligence and Machine Learning in Business Management Concepts, Challenges, and Case Studies, <https://doi.org/10.1201/9781003125129>, pp: 1–278.

9. Joshi S (2020) Sandeep Kumar Panda, Sathya AR, Optimal Deep Learning Model to Identify the Development of Pomegranate Fruit in Farms, *International Journal of Innovative Technology and Exploring, Engineering* 9(3):2352–2356
10. Sandeep Kumar Panda (2020) Vivek Bhalerao, Sathya AR, A Machine Learning Model to Identify Duplicate Questions in Social Media Forums, *International Journal of Innovative Technology and Exploring, Engineering* 9(4):370–373
11. Mahita Sri Arza, Sandeep Kumar Panda, An Integration of Blockchain and Machine Learning into the Health Care System,
12. Machine Learning Adoption in Blockchain-Based Intelligent Manufacturing, Vol 1, pp 33–58. DOI: <https://doi.org/10.1201/9781003252009-3>
13. Murala DK, Panda SK, Swain SK (2019) A survey on cloud computing security and privacy issues and challenges. *J. Adv. Res. Dyn. Control Syst.* 11:1276–1290
14. D.K. Murala, S.K. Panda, S.K. Swain: Secure dynamic groups data sharing with modified revocable attribute-based encryption in cloud. *Int. J. Recent Technol. Eng. (IJRTE)* 8(Issue4) (2019). ISSN: 2277–3878
15. Murala DK, Panda SK, Swain SK (2019) A novel hybrid approach for providing data security and privacy from malicious attacks in the cloud environment. *J. Adv. Res. Dyn. Control Syst.* 11:1291–1300
16. Panda SK, Satapathy SC (2021) Drug traceability and transparency in medical supply chain using blockchain for easing the process and creating trust between stakeholders and consumers. *Pers Ubiquit Comput.* <https://doi.org/10.1007/s00779-021-01588-3>
17. Hanumanthakari, S., Panda, S.K. (2022). Detecting Face Mask for Prevent COVID-19 Using Deep Learning: A Novel Approach. In: Satapathy, S.C., Bhateja, V., Favorskaya, M.N., Adilakshmi, T. (eds) *Smart Intelligent Computing and Applications, Volume 2. Smart Innovation, Systems and Technologies*, vol 283. Springer, Singapore. https://doi.org/10.1007/978-981-16-9705-0_45
18. Panda S.K., Satapathy S.C. (2021) An Investigation into Smart Contract Deployment on Ethereum Platform Using Web3.js and Solidity Using Blockchain. In: Bhateja V., Satapathy S.C., Travieso-González C.M., Aradhya V.N.M. (eds) *Data Engineering and Intelligent Computing. Advances in Intelligent Systems and Computing*, vol 1. Springer, Singapore. https://doi.org/10.1007/978-981-16-0171-2_52
19. Panda S.K., Rao D.C., Satapathy S.C. (2021) An Investigation into the Usability of Blockchain Technology in Internet of Things. In: Bhateja V., Satapathy S.C., Travieso-González C.M., Aradhya V.N.M. (eds) *Data Engineering and Intelligent Computing. Advances in Intelligent Systems and Computing*, vol 1. Springer, Singapore. https://doi.org/10.1007/978-981-16-0171-2_53
20. Panda S.K., Dash S.P., Jena A.K. (2021) Optimization of Block Query Response Using Evolutionary Algorithm. In: Bhateja V., Satapathy S.C., Travieso-González C.M., Aradhya V.N.M. (eds) *Data Engineering and Intelligent Computing. Advances in Intelligent Systems and Computing*, vol 1. Springer, Singapore. https://doi.org/10.1007/978-981-16-0171-2_54
21. Nanda, S.K., Panda, S.K., Das, M., Satapathy, S.C. (2022). Automating Vehicle Insurance Process Using Smart Contract and Ethereum. In: Chakravarthy, V.V.S.S.S., Flores-Fuentes, W., Bhateja, V., Biswal, B. (eds) *Advances in Micro-Electronics, Embedded Systems and IoT. Lecture Notes in Electrical Engineering*, vol 838. Springer, Singapore. https://doi.org/10.1007/978-981-16-8550-7_23
22. Varaprasada Rao, K., Panda, S.K. (2023). Secure Electronic Voting (E-voting) System Based on Blockchain on Various Platforms. In: Satapathy, S.C., Lin, J.C.W., Wee, L.K., Bhateja, V., Rajesh, T.M. (eds) *Computer Communication, Networking and IoT. Lecture Notes in Networks and Systems*, vol 459. Springer, Singapore. https://doi.org/10.1007/978-981-19-1976-3_18
23. Varaprasada Rao, K., Panda, S.K. (2023). A Design Model of Copyright Protection System Based on Distributed Ledger Technology. In: Satapathy, S.C., Lin, J.C.W., Wee, L.K., Bhateja, V., Rajesh, T.M. (eds) *Computer Communication, Networking and IoT. Lecture Notes in Networks and Systems*, vol 459. Springer, Singapore. https://doi.org/10.1007/978-981-19-1976-3_17

24. Panda, S.K., Elngar, A.A., Balas, V.E., & Kayed, M. (Eds.). (2020). *Bitcoin and Blockchain: History and Current Applications* (1st ed.). CRC Press. <https://doi.org/10.1201/9781003032588>
25. *Blockchain Technology: Applications and Challenges*, Editors: Panda, S.K., Jena, A.K., Swain, S.K., Satapathy, S.C. (Eds.), Springer, Intelligent Systems Reference Library. <https://doi.org/10.1007/978-3-030-69395-4>.
26. Sathya A.R., Panda S.K., Hanumanthakari S. (2021) Enabling Smart Education System Using Blockchain Technology. In: Panda S.K., Jena A.K., Swain S.K., Satapathy S.C. (eds) *Blockchain Technology: Applications and Challenges*. Intelligent Systems Reference Library, vol 203. Springer, Cham. https://doi.org/10.1007/978-3-030-69395-4_10
27. Lokre S.S., Naman V., Priya S., Panda S.K. (2021) Gun Tracking System Using Blockchain Technology. In: Panda S.K., Jena A.K., Swain S.K., Satapathy S.C. (eds) *Blockchain Technology: Applications and Challenges*. Intelligent Systems Reference Library, vol 203. Springer, Cham. https://doi.org/10.1007/978-3-030-69395-4_16
28. Sandeep Kumar Panda, Shanmukhi Priya Daliyet, Shagun S. Lokre, Vihans Naman, Distributed Ledger Technology in the Construction Industry Using Corda, *The New Advanced Society: Artificial Intelligence and Industrial Internet of Things Paradigm*, <https://doi.org/10.1002/9781119884392.ch2>
29. Panda SK, Mohammad GB, Nandan Mohanty S, Sahoo S. Smart contract-based land registry system to reduce frauds and time delay. *Security and Privacy*. 2021; e172. <https://doi.org/10.1002/spy2.172>