



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 5, May 2023

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

Impact Factor: 8.423

9940 572 462

6381 907 438

[ijrset@gmail.com](mailto:ijrset@gmail.com)

[www.ijrset.com](http://www.ijrset.com)

# Age, Gender and Mask Prediction using Deep Learning

Ms.Mahima M Gowda<sup>1</sup>, Pratheeksha R Nayak<sup>2</sup>, Sanjana P<sup>3</sup>, Sinchana Y M<sup>4</sup>, Suman D R<sup>5</sup>

<sup>1</sup> Assistant Professor, Department of CSE, ATMECE, Mysuru, India

<sup>2,3,4,5</sup> Student, Dept of CSE, ATMECE, Mysuru, India

**ABSTRACT:** Prediction of age, gender and mask from face images has drawn a lot of attention recently, due to its wide applications in various facial analysis problems. However, due to the large intra-class variation of face images (such as variation in lighting, pose, scale, occlusion), the existing models are still behind the desired accuracy level, which is necessary for the use of these models in real-world applications. In this work, we propose a deep learning framework, based on the ensemble of attentional and residual convolutional networks, to predict gender and age group of facial images with high accuracy rate. Using some mechanism enables our model to focus on the important and informative parts of the face, which can help it to make a more accurate prediction. We train our model in a multi-task learning fashion, and augment the feature embedding of the age classifier, with the predicted gender, and show that doing so can further increase the accuracy of age prediction. Our model is trained on a popular face age and gender dataset.

**KEYWORDS:** CNN, Image, Age estimation, Gender prediction Deep Learning

## I. INTRODUCTION

Age, gender and mask prediction has become one of the more recognized fields in deep learning, due to the increased rate of image uploads on the internet in today's data driven world. Humans are inherently good at determining one's gender, recognizing each other and making judgement about ethnicity but age estimation still remains a formidable problem. The Age and Gender Prediction is a classification project where we have to analyze an image and predict the gender and age. This is a deep learning project where we use image classification and regression models to obtain the results. Face recognition has emerged as one of the most difficult and intriguing problems in pattern recognition and image processing. With the aid of such a technology, one can easily detect a person's face by using a dataset of identical matching appearance. This method is useful in a variety of fields, including the military, defense, schools, colleges, and universities, airlines, banks, online web apps, gaming, and so on. Face masks are now widely used as part of standard virus-prevention measures, especially during the Covid-19 virus outbreak.

Many individuals or organizations must be able to distinguish whether or not people are wearing face masks in a given location or time. This data's requirements should be very real-time and automated. The challenging issue which can be mentioned in face detection is inherent diversity in faces such as shape, texture, color, got a beard moustache and/or glasses and even masks. From the experiments it is clear that the proposed CNN algorithm is very efficient and accurate in determining the facial recognition and detection of individuals.

## II. COMPUTER VISION

Computer vision assists computers by allowing them to look at/see, figure out, and identify digital images and movies in the same way that humans do. The difficulties it experiences are primarily due to biological vision and the advantage humans have over machines as a result of millions of years of development our civilization has through. This is something the computer struggles with since it fails to comprehend biological vision. Computer vision is primarily concerned with capturing, processing, analyzing, and comprehending digital images in order to extract data from the actual environment and provide symbolic or numerical information for decision-making. Object recognition, video tracking, motion estimates, and image restoration are all part of this process.

### III. DEEP LEARNING

Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces. Most modern deep learning models are based on artificial neural networks, specifically convolutional neural networks (CNNs), although they can also include propositional formulas or latent variables organized layer-wise in deep generative models such as the nodes in deep belief networks and deep Boltzmann machines.

### IV. METHODOLOGY

#### A. Convolutional Neural Network

A convolutional neural network (CNN) is a type of deep neural network (DNN) used for image recognition and processing, as well as natural language processing. A CNN, also known as a ConvNet, is a neural network that comprises input and output layers as well as several hidden layers, many of which are convolutional [5]. Convolutional neural networks have a great significance in image classification. The neural network cannot be fully connected for image recognition for a computer system or any model [6]. CNN are multilayer perceptrons that have been regularised. In the field of image classification, convolutional neural networks are quite useful. For picture recognition by a computer system or any model, the neural network cannot be fully coupled. The reason for this is that a picture often has a huge dimension and a large number of pixels, making fully connected neural networks difficult to employ because the number of weights required to train a model would be extremely large. For example, a 40\*40 pixel image would require 40\*40\*3 weights, or 4800 weights, which is still doable, but the image must be at least 200\*200 pixels in size or larger. However, utilising fully connected neural networks to recognise a 200\*200 image, the required.

#### Modle

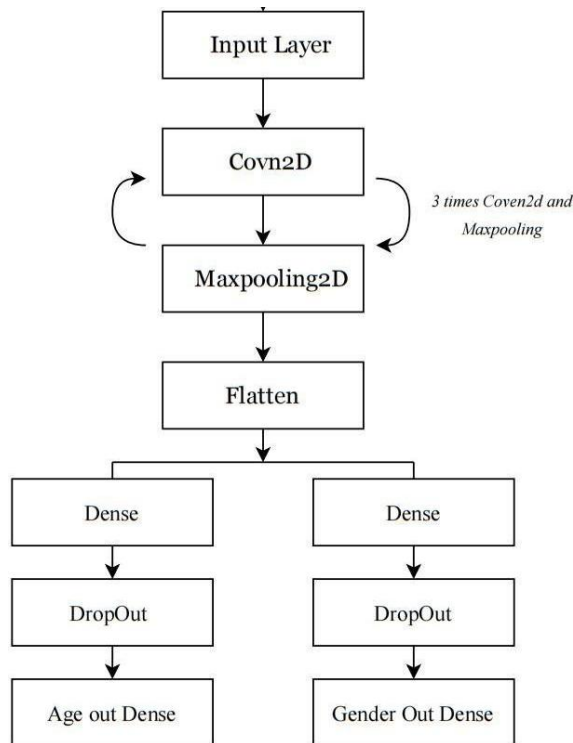


Fig 1: Architecture of the model

## 1. Input layer

The input layer is the first layer in a CNN. It receives the input image and passes it on to the next layer. The input layer is typically a 2D tensor, with dimensions of (height, width, channels). The number of channels depends on the number of color channels in the image. For example, a grayscale image has 1 channel, a color image has 3 channels (red, green, and blue), and a 3D image has 4 channels (red, green, blue, and depth).

## 2. Convolutional layers

Convolutional layers are the main building blocks of CNNs. They are responsible for extracting features from the input image. The convolutional layer takes the input image and applies a filter to it. The filter is a small matrix of weights that are learned during training. The filter is applied to the input image by sliding it across the image and computing the dot product of the filter and the input image at each location. The output of the convolutional layer is a feature map. A feature map is a 2D tensor that represents the features that have been extracted from the input image.

## 3. Pooling layers

Pooling layers are used to reduce the size of the feature maps. They do this by taking the maximum value or the average value of a small region of the feature map. The pooling layer helps to reduce the number of parameters in the model and to make the model more robust to changes in the input image.

## 4. Flatten layer

The flatten layer converts the feature maps into a 1D tensor. This is done so that the feature maps can be fed into the fully connected layers.

## 5. Fully connected layers

The fully connected layers are the last layers in a CNN. They take the output of the flatten layer and classify the image. The fully connected layers are similar to the neurons in a traditional neural network. They learn to classify the image by connecting to the feature maps and learning the weights that best separate the classes.

## 6. Output layer

The output layer is the last layer in a CNN. It produces the classification of the image. The output layer is typically a 1D tensor with one neuron for each class. The output of the output layer is a probability for each class. The class with the highest probability is the class that the image is classified as.

## B. Libraries

### 6. OpenCV

OpenCV stands for Open Source Computer Vision and is a free computer vision and machine learning library [5]. This package is mostly used for real-time picture and video preprocessing, as well as advanced analytics. It also works with a variety of Deep Learning frameworks, including TensorFlow, Caffe, and PyTorch. OpenCV is organised in such a way that it comprises a large number of shared or static libraries.

### 7. TensorFlow

Google designed and released TensorFlow, a Python library for rapid numerical computing. It is a foundation library that is mostly used to directly develop deep learning models, while wrapper libraries are occasionally used to facilitate the process that is built on top of TensorFlow.



## 8. Keras

Keras is a high-level Tensorflow Application Programming Interface (API) that provides the essential abstraction and major building blocks for the creation and distribution of machine learning solutions, as well as a very fast iteration rate. Keras assists engineers and researchers by encouraging them to fully utilise Tensorflow's scalability as well as its cross-platform capacity, which would not have been achievable without Keras. TPU may also be run on a huge cluster of GPUs, and you can export the Keras models to run in Google Chrome or any other browser, as well as on any smartphone or mobile device.



Fig 2: Sample images with face mask.



Fig 3: Sample images without face mask.

## C. Modules

### 1. Pre-processing Images

The input image is captured from a webcam or camera in real-time world. The frames (images) from the dataset are loaded. Face images are cropped and resized after they have been loaded. Later, noise distortions in the images are suppressed. Normalization is then done to normalize the images from 0-255 to 0-1 range.

### 2. Capture image ( )

In this Module it is able to capture real time images. This can be done by the help of Flutter and applying in to the Classifier Model. Input: Nothing.

### 3. Upload image ( )

Here browsing the image and upload for finding the Plant disease are possible. Firstly the image will be fetched and this image passes to Classifier Module. Input: Nothing Output: Image.

### 4. Classifier image ( )

Following data Preprocessing of the images, will apply to the Classifier. Here it will find out the feature of the images. Mainly in this module feature extraction occurs. Image similarity features will be stored into the model which gets created. Input: Image Output: Model.

### 5. Prediction image ( )

In this Module prediction of person take place. Here the browsed image will be placed in to the model and output will be shown as based on which label its get matched the most. Input: Image Output: PredictedLabel.



Fig 4: Sample images from the UTKFace dataset.

## V. MODEL TRAINING

### A. Age and Gender

The model starts with a series of convolutional layers (Conv2D) that extract features from the input image. The first layer has 32 filters of size 3x3, followed by a max pooling layer (MaxPooling2D) that reduces the size of the feature maps. This is repeated for the next two layers, with 64 and 128 filters respectively. Finally, there is a fourth layer with 256 filters, followed by another max pooling layer.

The output from the convolutional layers is then flattened to a one-dimensional vector using a Flatten layer. The flattened vector is then fed into two fully connected layers (Dense) with 256 units each, followed by a dropout layer with a dropout rate of 0.3 to prevent overfitting.

The two output branches each consist of a single fully connected layer with a different activation function. The first output branch uses a sigmoid activation function for binary classification (gender), while the second output branch uses a rectified linear unit (ReLU) activation function for regression (age).

The model is compiled with the binary cross-entropy loss function for the gender output and mean absolute error (MAE) for the age output. The Adam optimizer is used to minimize the loss, and the accuracy metric is used to evaluate the performance of the model during training.



```
Epoch 21/30
593/593 [#####] - 12s 20m/step - loss: 4.7885 - gender_val_loss: 0.1400 - age_val_loss: 4.5648 - gender_val_accuracy: 0.9487 - age_val_accuracy: 0.8657 - val_loss: 0.7990 - val_gender_val_loss: 0.3003 - val_age_val_loss: 6.5136 - val_gender_val_accuracy: 0.8943 - val_age_val_accuracy: 0.8059
Epoch 22/30
593/593 [#####] - 12s 20m/step - loss: 4.5717 - gender_val_loss: 0.1240 - age_val_loss: 4.4277 - gender_val_accuracy: 0.9458 - age_val_accuracy: 0.8672 - val_loss: 0.7676 - val_gender_val_loss: 0.3256 - val_age_val_loss: 6.4620 - val_gender_val_accuracy: 0.8991 - val_age_val_accuracy: 0.8076
Epoch 23/30
593/593 [#####] - 12s 20m/step - loss: 4.5410 - gender_val_loss: 0.1293 - age_val_loss: 4.4122 - gender_val_accuracy: 0.9463 - age_val_accuracy: 0.8694 - val_loss: 0.8375 - val_gender_val_loss: 0.3038 - val_age_val_loss: 6.5137 - val_gender_val_accuracy: 0.8785 - val_age_val_accuracy: 0.8086
Epoch 24/30
593/593 [#####] - 12s 20m/step - loss: 4.3856 - gender_val_loss: 0.1199 - age_val_loss: 4.2657 - gender_val_accuracy: 0.9480 - age_val_accuracy: 0.8687 - val_loss: 0.9885 - val_gender_val_loss: 0.3353 - val_age_val_loss: 6.5649 - val_gender_val_accuracy: 0.8842 - val_age_val_accuracy: 0.8072
Epoch 25/30
593/593 [#####] - 12s 20m/step - loss: 4.2792 - gender_val_loss: 0.1188 - age_val_loss: 4.1514 - gender_val_accuracy: 0.9478 - age_val_accuracy: 0.8618 - val_loss: 0.6233 - val_gender_val_loss: 0.3529 - val_age_val_loss: 6.4773 - val_gender_val_accuracy: 0.8914 - val_age_val_accuracy: 0.8221
Epoch 26/30
593/593 [#####] - 12s 20m/step - loss: 4.2451 - gender_val_loss: 0.1045 - age_val_loss: 4.1488 - gender_val_accuracy: 0.9548 - age_val_accuracy: 0.8628 - val_loss: 0.6227 - val_gender_val_loss: 0.3506 - val_age_val_loss: 6.5641 - val_gender_val_accuracy: 0.8908 - val_age_val_accuracy: 0.8266
Epoch 27/30
593/593 [#####] - 12s 20m/step - loss: 4.2041 - gender_val_loss: 0.1049 - age_val_loss: 3.9952 - gender_val_accuracy: 0.9541 - age_val_accuracy: 0.8527 - val_loss: 0.7162 - val_gender_val_loss: 0.3308 - val_age_val_loss: 6.8581 - val_gender_val_accuracy: 0.8876 - val_age_val_accuracy: 0.8248
Epoch 28/30
593/593 [#####] - 12s 20m/step - loss: 4.2119 - gender_val_loss: 0.1047 - age_val_loss: 4.0092 - gender_val_accuracy: 0.9529 - age_val_accuracy: 0.8335 - val_loss: 0.7088 - val_gender_val_loss: 0.3524 - val_age_val_loss: 6.4529 - val_gender_val_accuracy: 0.8888 - val_age_val_accuracy: 0.8237
Epoch 29/30
593/593 [#####] - 12s 20m/step - loss: 3.9985 - gender_val_loss: 0.0912 - age_val_loss: 3.8874 - gender_val_accuracy: 0.9588 - age_val_accuracy: 0.8326 - val_loss: 0.6278 - val_gender_val_loss: 0.3742 - val_age_val_loss: 6.4530 - val_gender_val_accuracy: 0.8914 - val_age_val_accuracy: 0.8278
Epoch 30/30
593/593 [#####] - 12s 20m/step - loss: 3.9830 - gender_val_loss: 0.0912 - age_val_loss: 3.8819 - gender_val_accuracy: 0.9599 - age_val_accuracy: 0.8365 - val_loss: 0.6272 - val_gender_val_loss: 0.3867 - val_age_val_loss: 6.5486 - val_gender_val_accuracy: 0.8925 - val_age_val_accuracy: 0.8386
```

Fig 5: We have performed 30 iterations and we have achieved accuracy for gender up to 95% and for age we have with you up to 89%.

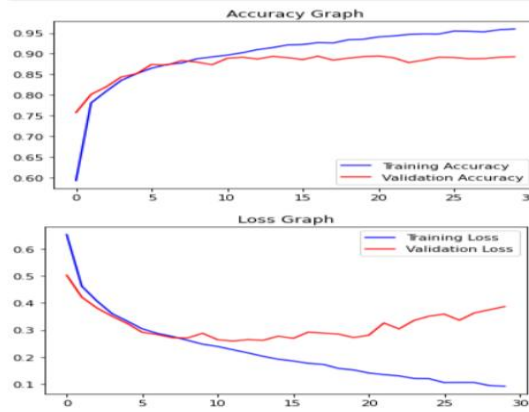


Fig 6: Accuracy Graph for Age and Gender

## B. Mask and NoMask

The script begins by defining a function `get_cnn_mode` that takes two arguments: `type` and `image_shape`. This function returns the pre-processing function and the CNN model to be used based on the `type` argument, which can be one of four CNN types: VGGNet, GoogleNet, ResNet, and MobileNet. The `image_shape` argument specifies the size of the input images to the model. The function uses the pre-trained models available in the Keras library and initializes the model with the weights pre-trained on the ImageNet dataset. It then returns the pre-processing function and the base model.

Next, the script initializes some hyperparameters like the initial learning rate, number of epochs to train for, and batch size. It then defines the input path and the list of classes to be classified: "WithMask" and "WithoutMask". It loads the images from the input path, resizes them to the specified image shape, preprocesses them using the pre-processing function returned by `get_cnn_mode` and appends them to the list of images. It also appends the corresponding label to the list of labels.

The script performs one-hot encoding on the labels and converts the images and labels to numpy arrays. It then splits the dataset into training and testing sets using a 80:20 split and constructs an image data generator for data augmentation.

The script then constructs the head of the model that will be placed on top of the base model. It uses the output of the base model as input to a global average pooling layer, which averages the spatial dimensions of the input tensor. It then flattens the tensor and passes it through two fully connected layers. The last layer uses a softmax activation function to output the probability of each class.

The script then sets the layers of the base model to be non-trainable and compiles the model using the binary cross-entropy

loss function, Adam optimizer with aspecified learning rate, and accuracy as a metric.

Finally, the script trains the model on the training data using the fit\_generator method of the model with the specified number of epochs and batch size. It also evaluates the performance of the model on the testing data using the evaluate\_generator method and prints the loss and accuracy on the test data.

```

Epoch: 0/100 [=====] - 18s 368ms/step - loss: 0.0013 - accuracy: 1.0000 - val_loss: 0.0522 - val_accuracy: 0.9875
Epoch 90/100
50/50 [=====] - 19s 371ms/step - loss: 0.0014 - accuracy: 1.0000 - val_loss: 0.0513 - val_accuracy: 0.9875
Epoch 91/100
50/50 [=====] - 19s 375ms/step - loss: 0.9323e-04 - accuracy: 1.0000 - val_loss: 0.0567 - val_accuracy: 0.9875
Epoch 92/100
50/50 [=====] - 19s 375ms/step - loss: 0.0017 - accuracy: 0.9994 - val_loss: 0.0563 - val_accuracy: 0.9875
Epoch 93/100
50/50 [=====] - 19s 370ms/step - loss: 0.0019 - accuracy: 0.9994 - val_loss: 0.0535 - val_accuracy: 0.9875
Epoch 94/100
50/50 [=====] - 19s 370ms/step - loss: 0.0028 - accuracy: 0.9987 - val_loss: 0.0535 - val_accuracy: 0.9875
Epoch 95/100
50/50 [=====] - 19s 372ms/step - loss: 0.0013 - accuracy: 1.0000 - val_loss: 0.0564 - val_accuracy: 0.9875
Epoch 96/100
50/50 [=====] - 19s 370ms/step - loss: 7.7607e-04 - accuracy: 1.0000 - val_loss: 0.0597 - val_accuracy: 0.9875
Epoch 97/100
50/50 [=====] - 18s 368ms/step - loss: 0.0025 - accuracy: 0.9994 - val_loss: 0.0595 - val_accuracy: 0.9875
Epoch 98/100
50/50 [=====] - 18s 368ms/step - loss: 7.4884e-04 - accuracy: 1.0000 - val_loss: 0.0594 - val_accuracy: 0.9875
Epoch 99/100
50/50 [=====] - 19s 370ms/step - loss: 0.0080 - accuracy: 0.9994 - val_loss: 0.0580 - val_accuracy: 0.9875
Epoch 100/100
50/50 [=====] - 18s 367ms/step - loss: 0.0014 - accuracy: 1.0000 - val_loss: 0.0628 - val_accuracy: 0.9875
    
```

Fig 7: We have performed 50 iteration and we have achieved accuracy for gender up to 98%.

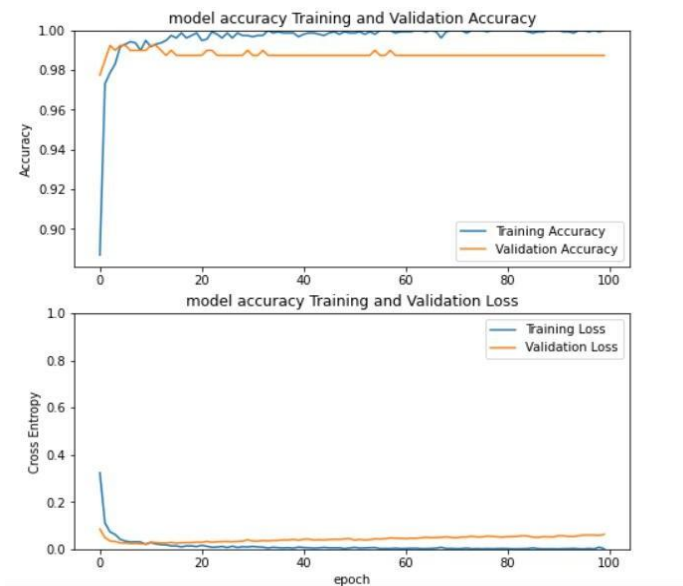


Fig 8: Accuracy Graph for Mask

## VI. RESULT

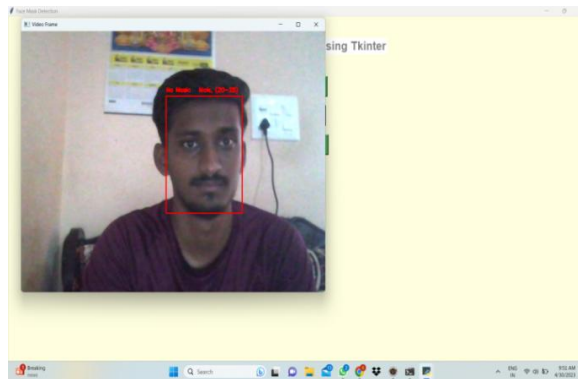


Fig 7: Age, Gender and Mask Prediction.



## VII.CONCLUSION

Identifying the age and gender of the individuals we come across in our daily lives has an important role in our social lives too. For example, languages used to salute for men and women are very often different, and the words used to address the elders and the young are different too. We human beings are able to evaluate the individual's age and gender just from their facial appearances. Nowadays, many smart applications that include visual surveillance, medical diagnosis, and marketing intelligence, need to evaluate the same for individuals using their facial images. Here lies the importance of a robust and efficient methodology for gender and age estimation through the computing devices. Deep Learning and convolutional neural networks will be used to reliably determine a person's gender and age range from a single photograph of a face in the future using this Prototype. The image may be from a certain dataset, could be parsed in a specific way by the prototype, or it could even be real-time if no arguments are processed.

## VII.FUTURE SCOPE

During public health crises, such as pandemics, deep learning models can be employed to monitor and ensure compliance with social distancing measures. Predicting age, gender, and mask usage can help authorities identify crowded areas, enforce distancing rules, and take appropriate actions to minimize the risk of spreading infections. Age and gender prediction can be used in healthcare settings for patient management, personalized medicine, and clinical research. By analyzing these factors, medical professionals can tailor treatments, predict health risks, and develop targeted interventions. Deep learning models that predict age, gender, and mask usage can assist marketers in delivering personalized advertisements and product recommendations. This targeted approach can improve customer engagement, conversion rates, and overall marketing effectiveness. Deep learning models trained on specific datasets might struggle to generalize to different settings, lighting conditions, or ethnicities. Future work can focus on creating more diverse and representative datasets and developing robust models that can handle variations across different populations and environments.

## REFERENCES

- [1] C. Szegedy, W. Liu Y. Jia et al USA, "Going deeper with convolutions". IEEE Conference on (pp.112-119).IEEE,June2015.
- [2] Qawaqneh,Mallouh, and Barkana .,"Deep neural network framework and transformed MFCCs for speaker's age and gender classification" IEEE Transactions on Evolutionary Computation, vol.4, no.2, pp. 164–171,2019.
- [3] Gil Levi et al,"Gender ,age and ethnicity detection", in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp.714–721, Las Vegas,NV,USA,June2016.
- [4] M.DelCoco et al,"Gender-dependent differences in a cohort of Charcot-Marie-tooth (CMT) patients," International Journal of Advanced Research in Computer Engineering & Technology, vol. 1,no. 8, pp. 923–925,2018.
- [5] Grassmann, Felix, Judith Mengelkamp, Caroline Brandl, Sebastian Harsch, Martina E. Zimmermann, Birgit Linkohr, Annette Peters, Iris M. Heid, Christoph Palm, and Bernhard HF Weber. "A deep learning algorithm for prediction of age-related eye disease study severity scale for age-related macular degeneration from color fundus photography." *Ophthalmology* 125, no. 9 (2018): 1410-1420.
- [6] A. Patel, O. V. Swathika, U. Subramaniam et al., "A practical approach for predicting power in a small-scale off-grid photovoltaic system using machine learning algorithms," *International Journal of Photoenergy*, vol. 2022, 21 pages, 2022.
- [7] A. Hmidet, U. Subramaniam, R. M. Elavarasan et al., "Design of efficient off-grid solar photovoltaic water pumping system based on improved fractional open circuit voltage MPPT technique," *International Journal of Photoenergy*, vol. 2021, Article ID 4925433, 18 pages, 2021.
- [8] R. Nanmaran, S. Srimathi, G. Yamuna et al., "Investigating the role of image fusion in brain tumor classification models based on machine learning algorithm for personalized medicine," *Computational and Mathematical Methods in Medicine*, vol. 2022, 13 pages, 2022.
- [9] M. Mandal, P. K. Singh, M. F. Ijaz, J. Shafi, and R. Sarkar, "A tri-stage wrapper-filter feature selection framework for disease classification," *Sensors*, vol. 21, no. 16, p. 5571, 2021.



- [10] Rothe, R., Timofte, R., & Gool, L. V. (2015). Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126(2-4), 144-157.
- [11] Antipov, G., Baccouche, M., & Dugelay, J. L. (2017). Face aging with conditional generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4310-4319).
- [12] Wang, N., Gao, X., Tao, D., & Li, X. (2016). Aging face synthesis with residual generative adversarial networks. In *European Conference on Computer Vision (ECCV)* (pp. 797-812).
- [13] Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2017). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10), 1499-1503.



**INNO SPACE**  
SJIF Scientific Journal Impact Factor  
**Impact Factor: 8.423**



**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

**9940 572 462** **6381 907 438** **ijirset@gmail.com**



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details