

# Multi-Task Learning for Age Group Estimation and Gender Recognition using Facial Features



Oluwole Charles Akinyokun<sup>1\*</sup>, Abejide Michael Adegoke<sup>1</sup> and Taye Oluwabusayo Akinrele<sup>2</sup>

<sup>1</sup>Bells University of Technology, Nigeria

<sup>2</sup>Federal University of Technology, Nigeria

**Submission:** February 14, 2022; **Published:** March 02, 2022

**\*Corresponding author:** Oluwole Charles Akinyokun, Bells University of Technology, Ota, Nigeria

## Abstract

Machine learning has played an important role in image analysis and computer vision especially in the estimation and prediction of age group and gender of people based on their facial features. Research, distinctive features on the human face such as the shape, size and texture, contribute largely to the estimation of age, also the position and shape of facial landmarks in the image of a person contribute to the recognition of gender. In this research, an attempt is made to compare the performance of two multi-task learning models, namely Convolutional Neural Network (CNN) and Support Vector Machine (SVM) classifiers for human age groups and gender. The research classified age into eight groups namely: (0-2), (4-6), (8-12), (15-20), (25-32), (38-43), (48-53) and (60-100), and gender into male and female. An experimental study using benchmark database images was carried out and the evaluation of the results obtained were compared and reported.

**Keywords:** Machine learning; Image analysis; Computer vision; Facial recognition; Convolutional Neural network; Support vector machine

**Abbreviation:** CNN: Convolutional Neural Network; SVM: Support Vector Machine; RELU: Rectified Linear Unit; LBP: Local binary patterns; FPLBP: Four Patch LBP; BHEP: Bilateral Histogram Equalization; ANN: Artificial Neural Network; PLS: Partial Least-Squares; SAM: Supervised Appearance Model

## Introduction

The human face conveys a lot of information which people have an astonishing ability to extract, analyze and decipher [1]. The main characteristic feature of the human being is the face which exhibits different emotions that can be determined and easily predicted by several facial expressions [2]. Just by glancing at the face of a person, one can estimate the age and gender of that person. Identifying human faces and modelling the distinguishing features contribute toward face recognition are some of the challenges faced by computer vision and psychophysics researchers [3]. Age and gender are significant properties regarded as crucial biological characteristic, which plays a fundamental role in human social interaction. The human face contains a wide range of information for gender perception and age estimation [4]. Classification of age and gender is an important visual task for human beings since many social interactions critically depend on the correct age and gender perception.

As technology such as visual surveillance and human computer interaction evolve, computer vision systems for age and gender

classification play an important role in ones live. It is therefore not surprising that a lot of research has been carried out to investigate age and gender classification from face perception in humans, proposing various methods for a machine to attain the human level of accuracy. Predictably, it is challenging for machines to identify visual information since discriminative feature extraction is easily affected by various factors like large variations in facial gestures, lighting, background and so on. Perceived age and gender classification are a topic with a high application potential in areas like surveillance, face recognition, video indexing and dynamic marketing surveys.

This research attempts to compare two multi-task learning models namely: Convolutional Neural Network (CNN) and Support Vector Machine (SVM), in terms of performance while classifying humans into age groups and gender. The proposed approach treats gender prediction and age estimation as a classification problem, adopting datasets that compose images labelled for age and gender. It uses eight classes of age group namely: (0-2), (4-

6), (8-13), (15-20), (25-32), (38-43), (48-53) and (60-100), and classifies gender into male (M) and female (F). The evaluation of these models peered with the current state-of-the-art methods and the efficiency of the models was demonstrated and compared through the experimental results using benchmark database images. The research aimed to design a web-based assessment system that will solve the problem of the collection of data from manual sources like mobile phones or Internet. The specific objectives of the research were to:

- a) Design a CNN model and SVM model to jointly recognize the gender and age group of a person through an input image.
- b) Carry out the comparative analysis of the CNN and SVM models using Audience image datasets.
- c) Design a web-based model evaluation system.

The motivation for this research was the observation that the amount of data available for the study of computer vision problems can have an enormous impact on the machine capabilities developed to solve them. The study made use of different datasets to train two multi-task learning models namely: CNN and SVM and compare their performance in other to generate an accurate result in the prediction of gender and age groups from input images. Some datasets that were used in the study include the IMDB-Wiki dataset (523,051) face images, Adience dataset (34,795) images and the UTK Face dataset (over 20,000) images. The study aimed at the design of a web site based assessment system that will solve the problem of the collection of data from manual sources like mobile phones or computers.

### Overview of CNN and SVM

In machine learning, classification is considered an instance of supervised learning where a training set of correctly identified observations is available. It involves computer program learning from the data input given to it and using this data to classify new observations [5]. Classification may be binary or multi-class where the former can be used to identify whether a person is male or female and the latter is used to identify the age group of a person. Some examples of classification problems are speech recognition, handwriting recognition, biometric identification, age recognition, document classification, gender prediction and so on. There are different types of classification algorithms in machine learning and they include Linear Classifiers such as Logistic Regression and Naïve Bayes Classifier, Nearest Neighbour, SVM, Decision Trees, Boosted Trees, Random Forest, and Neural Networks. In this research, CNN and SVM are reviewed in some details.

Machine learning is the practice of using algorithms to analyze data, learn from the data, and then make a determination or prediction about new data [6]. In contrast to the logical algorithm in which computers are explicitly instructed on how to accomplish a specific task, the machine learning approach obtains

the instruction for carrying out a task by learning patterns from data (Russel and Norvig, 2011). The goal of machine learning is to develop the ability to engage in human-like processes such as: learning, adapting, self-correction, driving, understanding natural languages, problem solving and use of data for complex processing tasks [7]. In artificial intelligence systems, the conditions for carrying out a task are not manually and explicitly defined by programmers but by an inductive process, otherwise called a learner, which observes or studies the characteristics of an existing data to compute the function for carrying out the task. That is, rather than manually coding a specific set of instructions to accomplish a task, the machine is trained, using data, to build the condition for accomplishing the task. This way the learners handle uncertainty that might not be envisaged by a programmer.

A machine learning model uses likelihood function to familiarize itself with the features of the existing data. This is called training [6], that is using existing data to acquire knowledge or skills for carrying out tasks. Machine learning models are trained for them to effectively carry out their respective tasks and the more data that are available to the models for training, the better and more robust the model will be [6]. The following are the three learning mechanisms these models can adopt, namely: Supervised learning, Unsupervised learning and Semi-Supervised learning. In the supervised learning, the model is guided on how to carry out a task by labeling the data used for training. That is, each piece of data that is passed to the model during training is a pair that consists of the input object or sample, along with the corresponding label or output value. The rules for accomplishing a similar task in the future are obtained from the labeled data. Though the rules for carrying out similar tasks are generated by the model, the learner becomes perplexed and possibly does a wrong thing if it encounters strange data [8,9] that is a data that does not have the features of the data that were used to train the model. That is why it is necessary for the model to be trained with as many data as can be envisage. If there are no large dataset to train with, the scope of the model will be limited. It is however a time-consuming process to manually create such a large number of training data, even when they are available. Hence, the invention of semi-supervised learning.

Semi-supervised learning occurs when there is availability of large dataset for training in which manually labeling all these data is impracticable. Some portions of the large dataset are labeled and use to train the model using a technique called pseudo labeling. Pseudo-labelling involves the following processes:

- a) Manually label some portions of the data
- b) Use the labeled data as the training set and train the model with the labeled dataset
- c) Use the model to predict the remaining unlabeled portion of data

d) Take these predictions and labeled each piece of unlabeled data with the outputs that is predicted for the individual data.

Pseudo labeling thus allows training on a vastly larger dataset that otherwise may have taken many tedious hours of human labour to accomplish. In the unsupervised learning, the data for training the model are not labeled but the model, on its own, learn structures from the data and extract useful information or features that the model can use to carry out its future tasks.

Unsupervised learning mechanism analyses data, use some features extracted from the data to put similar data together, and then generates rules for processing future data. It is indeed a clustering model. In unsupervised learning, the labels and the rules to process a future data are learned and generated automatically from the underlying data (Jeffrey, 2007) [10,11]. Examples of techniques used for generating the labels automatically or clustering data include single-word indexing technique [12], Latent semantic indexing [13], Probabilistic latent semantic indexing (PLSI) [14], Latent Dirichlet allocation [15], Data-driven Dirichlet allocation [10,11]. All of these models use one optimization technique or the other for training. Examples of machine learning models that uses data to gain knowledge or skills that will enable them to effectively carry out tasks include Artificial Neural Networks, Bayesian Networks, Support Vector Machines (SVM).

### Overview of CNN

A neural network consists of units (neurons), arranged in

layers, which converts an input vector into some output vectors. Each unit takes an input, applies a function often nonlinear to it and passes the output on to the next layer [5]. This method is one of the most popular classification methods used in many domains and the method was proposed by many authors as a classifier in age and gender recognition cases. Before an input image is used in CNN, it is scaled to a chosen size. The size of the image determines the number of network input nodes so that the image will be as small as possible, but its size will be enough to keep the important information about the facial features. Each input image will travel through a succession of convolution layers with filters (kernels), pooling and fully connected layers to train and evaluate deep learning CNN model and apply softmax function to classify an object with probabilistic values between 0 and 1 [16].

**Convolution layer:** It is the first layer to extract features from an input image. It preserves the relationship between pixels by learning image features using squares of input data. This layer is a mathematical operation that takes two inputs such as an image matrix and a filter or kernel. The Figure 1 below illustrates what this layer does.

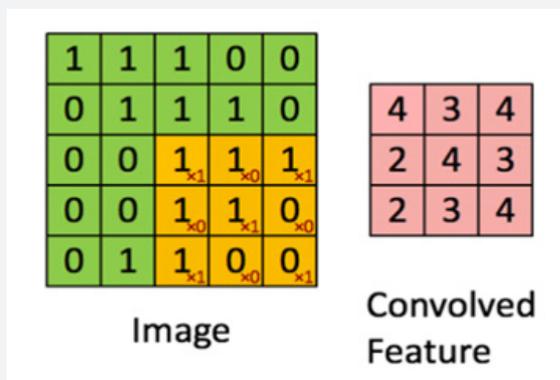
$$Z = X * f; \text{ where } X = \text{the input image and } f = \text{the kernel (filter)}$$

Asterisksign (\*) represents convolution

Dimension of image (X) = (n, n)

Dimension of filter (f) = (f, f)

Dimension of output (Z) = ((n-f+1), (n-f+1))



**Figure 1:** Convoluting a 5x5x1 image with a 3x3x1 kernel to get a 3x3x1 convolved feature.

Some key terms in this layer are stride, padding and ReLU. Stride is the number of pixels shifts over the input matrix [16], Padding means adding pixels of some value around the input images, it is used when the filter does not fit perfectly into the input image, it is of two types (same padding or valid padding) [17], Rectified Linear Unit (RELU) is an activation function, its purpose is to introduce non-linearity in the ConvNet since the

real-world data would want the ConvNet to learn non-negative linear values.

**Pooling layer:** It is responsible for decreasing the special size of the convolved feature to decrease the computational power required to process the data through dimensionality reduction. It is also useful for extracting dominant features which are rotational and positional invariant, thereby maintaining the

process of effective training the model. They are of two types: Max Pooling and Average Pooling, the former returns the maximum value from the portion of the image covered by the kernel, while

the latter returns the average of all the values from the portion of the image covered by the kernel [17]. These terms are illustrated in the Figure 2 below.

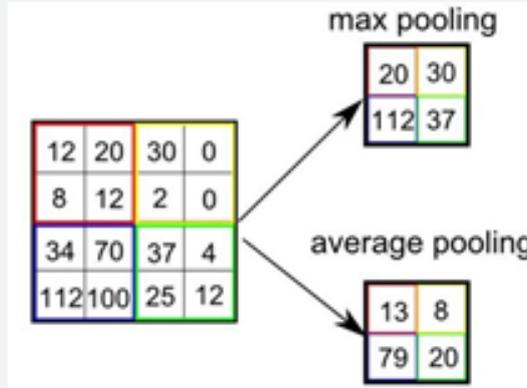


Figure 2: Different types of pooling.

**Fully connected (FC) layer:** In the FC layer, the matrix is flattened into a vector and fed into it like a neural network. Adding this layer is usually a cheap way of learning non-linear combinations of high-level features as represented by the output of the convolutional layer. Over a series of epochs, the model can distinguish between dominating and certain low-level features in images and classify them using the softmax classification technique. The following operations were performed on the convoluted feature:

a) Linear Transformation;  $Z = WT \cdot X + b$

$X$  = Input Image

$W$  = Weight

$b$  = bias

$$X = \begin{bmatrix} X1 \\ X2 \\ X3 \\ X4 \end{bmatrix} \quad W = \begin{bmatrix} W11 & W12 \\ W21 & W22 \\ W31 & W32 \\ W41 & W42 \end{bmatrix} \quad b = \begin{bmatrix} b1 \\ b2 \end{bmatrix}$$

Input Data
Randomly Initialized
Randomly Initialized  
Weight Matrix
bias matrix

$$Z = \begin{bmatrix} W11 & W21 & W31 & W41 \\ W21 & W22 & W32 & W42 \end{bmatrix} \begin{bmatrix} X1 \\ X2 \\ X3 \\ X4 \end{bmatrix} + \begin{bmatrix} b1 \\ b2 \end{bmatrix}$$

**b) Non-Linear Transformation:** Linear transformation alone cannot capture complex relationships thus, non-linearity transformation is added to the component in the network which is called the activation function. It is added to each layer in the neural network. The activation function used is dependent on the type of problem being solved. In this project, the softmax function is used. The formula is:

c)  $\sigma(\bar{Z})_i = \frac{e^{Z_i}}{\sum_{j=1}^k e^{Z_j}}$

$\sigma$  = softmax

$\bar{Z}$  = input vector

$e^{Z_i}$  = standard exponential function for input vector

$k$  = number of classes in the multi-class classifier

$e^{Z_j}$  = standard exponential function for output vector

The image in Figure 3 above represents the architecture of the CNN. CNN can learn relevant features from an image at different levels similar to the human brain, it has the main feature, weight sharing, therefore, it is more efficient in terms of memory and complexity. It is also an outstanding feature extractor especially when dealing with a completely new task or problem. CNN has many disadvantages in that it is computationally expensive and it tends to forget about a previously trained task if it wants to learn a new one, which is due to rewriting of connection efficacies or weight by the learning algorithm. Another disadvantage is that it is a supervised learning algorithm, this is a weakness because the main aim of machine learning is that they learn by itself.

**Overview of SVM**

Support Vector Machine (SVM) is a linear model for classification and regression problems. It can solve both linear and nonlinear problems and it is useful for a wide range of applications. The idea of SVM is straightforward: the algorithm creates a line or a hyperplane which separates the data into classes. The original SVM algorithm was invented by Vapnik and Chervonenkis in 1963. Figure 4 below describes how the SVM classifier works. The SVM can be defined as a learning algorithm for pattern classification and regression. The basic convention behind SVM is finding the optimal linear hyperplane such that the expected classification error for unseen test samples is minimized, that is,

good generalization performance. However, if the two classes are not linearly separable, the SVM attempts to find the hyperplane

that maximizes the margin at the same time, minimizing a quality proportional to the number of misclassification errors.

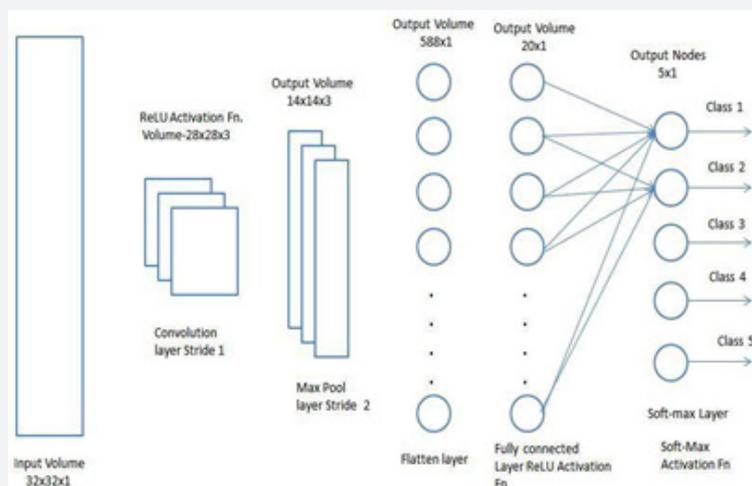


Figure 3: Complete CNN architecture.

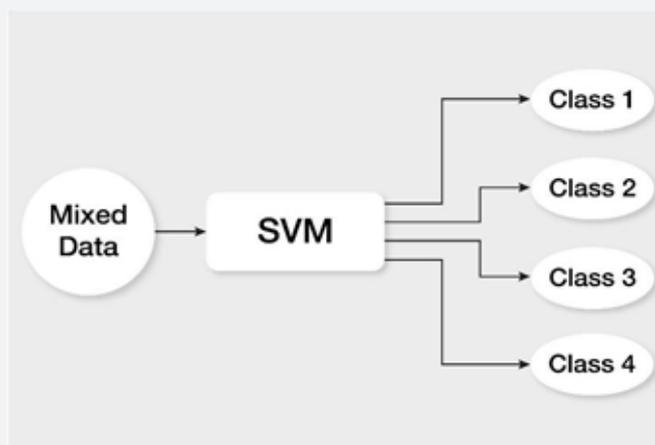


Figure 4: Support vector machine classifier.

The SVM training technique produces a model that assigns new examples to one of two categories, making it a non-probabilistic binary linear classifier, given a set of training examples, the Audience dataset, each tagged as belonging to one of two categories. More formally, the SVM constructs a hyperplane or a set of hyperplanes in a high or infinite dimensional space, which can be used for classification, regression and other tasks like outliers detection. A typical illustration is shown in Figure 5 below. Some advantages of SVM is effective in high dimension spaces, also in cases where the number of dimensions is greater than the number of samples, it is memory economical because it employs a subset of training points (called support vectors) in the decision function. Finally, it is versatile because different kernel

functions can be chosen for the decision function; common kernels are available, but custom kernels can also be defined. The SVM has several disadvantages which include, if the number of features is much greater than the number of samples, avoiding over-fitting in choosing kernel functions and regularization term is crucial, also, they do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

The CNN is non-linear classifier whereas SVM is a linear classifier. The CNN works well with visual image recognition while SVM is used widely in classification problems. The CNN increases model complexity by adding more layers but in SVM increasing model complexity is not possible.

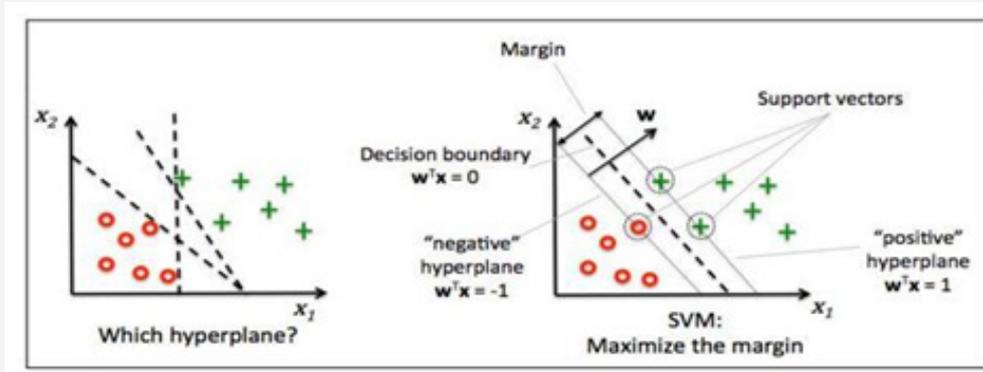


Figure 5: SVM Illustrating the hyperplane for data of two classes

### Literature Review of Related Works

The classification of the age and gender using facial features is an interesting field of research. It is therefore imperative to carry out the review of some popular existing research works in the following.

The age and gender estimation of unfiltered faces was carried out in [18]. The primary reason for the study was the conclusion that the amount of data available for the study of a computer vision problem can have an immense impact on the machine capabilities to solve it. The objective of the study was to narrow the differences between the efficiency of age and gender estimation and face recognition systems, mainly by providing the resources needed for the study of the former. The system used consists of detection, alignment and identification (representation and classification). Given a photo, the standard Viola and Jones face detector was first applied, then detected faces are aligned to a single reference coordinate frame. Aligned faces were encoded using several popular global image representations. Local binary patterns (LBP) and the related Four Patch LBP codes (FPLBP) were chosen and these were selected due to their successful application to face recognition problems, as well as their efficient computation and representation requirements. Classification was performed using standard linear SVM trained using the feature vector representations listed above. Each descriptor was examined independently or combine with multiple descriptors by concatenating them into single long feature vectors. Training was performed using dropout-SVM scheme and classification of gender is performed using a single linear SVM classifier. For the multi-label age classification, a one-vs-one linear-SVM arrangement was used. The choice of a simple linear-SVM was motivated by the desire to reduce the risk of over-fitting. Results showed that by training linear-SVM classifiers by dropout-SVM, excellent results were obtained without apparent over-fitting. The study provided age and gender benchmark of unfiltered face photos, robust

face alignment and dropout-SVM and performance evaluation of alternative benchmarks and the irrespective levels of difficulty as well as the capabilities of automatic age and gender estimation systems. The system was tested on two variants of the benchmark, as well as two variants of the Gallagher benchmark.

Age Group Estimation and Gender Recognition Using Face Features was attempted by [19]. The motivation behind that work was that there was no effective method to estimate the age group and gender of individuals. The study therefore provided a technique to estimate the age group and gender using face features. It provided a methodology to estimate age group and gender using face features which involved the following four stages:

**a) Pre-processing:** The face of a person was captured by a digital camera, this stage included three steps which were detecting the image, converting the image to grayscale and noise-reduced image. The colour image represented in binary digits 0 and 1 and the Gaussian filtering method was used for noise reduction.

**b) Face normalization:** In the normalization process, the system cropped the detected rectangular face area using Matlab in-built object function. Then eye pair, mouth, nose, and chin were detected which gave the specific images of the left eye, right eye, left eyebrow, mouth that were image of lips and also chin hairline part of the face was detected with the nose image.

**c) Feature extraction:** A combination of global and grid features were extracted from face images. The global features such as distance between two eyeballs, eye to nose tip, eye to chin, and eye to lip were calculated using four distance values. Four features F1, F2, F3 and F4, which denoted the global features and the feature F5 were calculated for grid features. The Canny edge detection technique was used for finding the grid features. The four features F1, F2, F3 and F4 were calculated as follows:

- i.  $F1 = (\text{distance from left to right eye ball}) / (\text{distance from eye to nose})$
- ii.  $F2 = (\text{distance from left to right eye ball}) / (\text{distance from eye to lip})$
- iii.  $F3 = (\text{distance from eye to nose}) / (\text{distance from eye to chin})$
- iv.  $F4 = (\text{distance from eye to nose}) / (\text{distance from eye to lip})$
- v.  $F6 = \text{the angle between the right eyeball, mouth point, and left eyeball in the face image}$

Using the Grid features of the face image, feature F5 was

calculated. It was entirely based on wrinkle geography in the face image. The grid feature included the forehead portion, eyelid regions, the upper portion of cheeks and eye colour regions. To calculate feature F5, the following steps had to be followed: The colour face image was converted into a grayscale image. Then canny edge detection technique was applied to the grayscale face image. It gave a binary face image with wrinkle edges.

**d) Classification:** Age ranges were classified dynamically depending on the number of groups based on the above six features F1 to F6. The SVM was used as the age classifier technique. The age classification was carried out in 2, 3 and 4 age range groups shown in below. Using five features F1 to F5, age classification was done into 5 age range groups (Table 2).

**Table 1:** Accuracy for age group.

S/N	Age Group	True Accuracy
1	5-17	90%
2	18-30	100%
3	31-50	85%
4	51-70	70%
5	71-100	85%

**Table 2:** Accuracy for gender.

S/N	Class	True Accuracy LBP	True Accuracy FPLBP	True Recognition Accuracy
1	Female	100%	80%	95%
2	Male	90%	70%	75%

The geometric features of facial images like wrinkle geography, face angle, left eye to right eye distance, eye to chin distance and eye to lip distance were calculated. Based on the texture and shape information, age classification was carried out, age ranges were classified dynamically depending on the number of groups using the SVM classifier algorithm. The study can be used for predicting future faces, classifying gender and expression detection from images. An android camera (SAMSUNG Galaxy on 8-SM-J710FN) was used to capture the face images of some persons, and some images were collected from net data set images. The study presented a novel and effective age-group estimation using face features from human face images.

Gender Classification and Age Detection Based on Human Facial Features Using Multi-Class SVM was carried out in [2]. In the human being, the key characteristic feature is the face. The various emotions of a human can be determined and easily realized by the different facial expressions. The face is considered to be the most acceptable biometric trait than any other component as image capturing and prediction of images is easier to perform than other traits. Faces are normally classified as semi-rigid, semi-flexible, culturally significant, part of our entity and thus need good computing techniques for face recognition and classification. The research attempted to perform gender classification of human

facial images based on the extracted feature from the input image set. A novel methodology was proposed to achieve the goal of gender classification and age detection in a three-step process. First, the input image set was pre-processed to perform noise removal, histogram equalization, size normalization and face detection. Second, feature extraction from the facial image was performed. Finally, evaluation of the performance of the proposed algorithm. Experiments were carried out on a various images set that contains an equal proportion of male and female by using suitable binary SVM classifier which will classify the data into two male or female. To achieve the second goal, Multiclass SVM was employed to generate three classes namely child, adult, and old. The age of the input images was detected and classified into one of the three categories. Figure 6 given below depicts these steps:

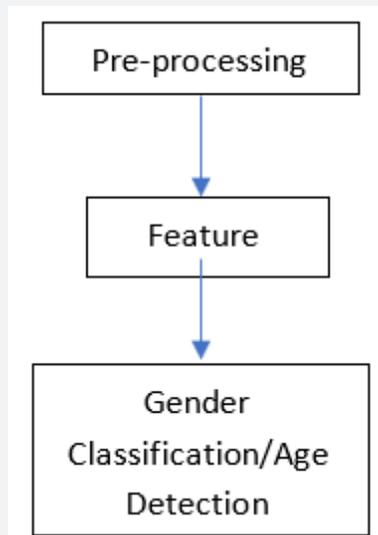
**Steps:**

**a) Pre-processing:** This step involved operations like image normalization which was a process of changing the intensity values of pixel range in an image; histogram equalization which improves the contrast of an image to stretch out the intensity range so that the area with the lower contrast can achieve higher contrast and noise removal which achieved by implementing different techniques like Linear Filtering or Median Filtering.

**b) Feature Extraction:** This was achieved by adopting two approaches which were geometry-based approach and template-based approach. While the former used geometric information such as features, relative positions and sizes of the face components as features measure the later was used to match with appropriate

energy function in the design of standard face pattern.

**c) Classification:** In this last step, the SVM classifier was used to classify the age and gender of the input images based on the facial features extracted.



**Figure 6:** Steps involved in gender and age classification.

The paper proposed a methodology for gender classification and age detection using multiclass SVM. The limitation of the study was the insufficient amount of database with facial images to be used in training and testing of the SVM classifier.

An approach for classifying gender from facial images using the feature extraction method centrist was proposed in [20]. A framework was proposed where the pre-processing to enhance the images using Bilateral Histogram Equalization (BHEP) algorithm was applied to mitigate the problems of the high rate of noises, lack of illumination and so on. Three different datasets were used in the study to evaluate the proposed system and then experimental results with experimental settings and detailed training and testing procedures were developed. The contributions of the study were as follows:

- a) Introduction of an automated system for gender recognition from facial images.
- b) It used Bilateral Histogram Equalization (BHEP) for enhancing the input image.
- c) It adopted centrist for extracting features from the input images.

CNN for Age and Gender Classification was proposed in [21]. The input to the algorithm developed was an image of a human face of size 256x256 that was cropped to 227x227 and fed into either the age classifier, gender classifier or both. The age classifier

returned an integer representing the age range of the individual thus (0-2, 4-6, 8-13, 15-20, 25-32, 38-43, 48-53, 60+) and the gender classifier returned a binary result where 0 represent male and 1 represents female. The objective of the research was to develop a system that leverages the inherent inter-relationships between age and gender to link the architectures in such a way that the overall performance would be improved. The network architecture developed was intended to be relatively shallow to prevent overfitting the data. An RGB image being input to the network was first scaled to 3x256x256 and then cropped to 3x227x227. There were three convolution layers which were completely connected as shown in Figure 7. The dataset was divided into six subject-exclusive folds, and further divided each of the folds into males and females, and each of the eight age groups. This resulted in a total of 66 "sub-folds", where each of the original six-folds was broken up into eleven groups, based on the types of classifiers that would be trained. The dataset used for training and testing for the study was the Audience face dataset, which comes from the Face Image Project [22] from the Open University of Israel (OUI). The dataset contains a total of 26,580 photos of 2,284 unique subjects that were collected from Flickr. Each image was annotated with the person's gender and age range (out of 8 possible ranges). The images were subject to various levels of occlusion, lighting and blur, which reflect real-world circumstances. The author used those images which were mostly front-facing, which limited the total number of images to around

20,000. The Figure 7 showed some examples of images of both males and females in the dataset of various ages. The images were

original of size 768x768, so they were preprocessed by all being resized down to 256x256.



**Figure 7:** Audience image dataset examples. top row: 6 males of various ages. bottom row: 6 females of various ages.

The most difficult portion of the study was setting up the training infrastructure to properly divide the data into folds, train each classifier, cross-validate and combine the resulting classifiers into a test-ready classifier. The author foresaw future directions building off the work to include using gender and age classification to aid face recognition, improve experience with photos on social media and much more. It was also proposed that training separate age classifiers for men and women could simulate the added power of deeper networks while avoiding the danger of over-fitting. The limitation of the study was the inability to have access to the personal information of the subjects in the images, namely, date of birth and gender, stating that the date of birth of subjects was a rarely released piece of information.

Age and Gender Classification using Convolutional Neural Networks was developed in [23]. The performance of existing methods on real-world images which were unquestionably lacking especially when compared to the tremendous leaps in performance recently reported for the related task of face recognition was the primary motivation for the study. The goal of the study was to close the gap between automatic face recognition capabilities and those of age and classification methods. Obtaining a large, labelled image training set for age and gender estimation from social image repositories necessitates either access to personal information on the subjects appearing in the images (their birth date and gender), which was often restricted or manually labeling the images, which was tedious and time-consuming. Datasets for age and gender estimation from real-world social images were therefore relatively limited in size and presently no match in size with the much larger image classification datasets. Over-fitting was a common problem when machine learning-based methods were used on such small image collections. This problem was exacerbated when considering deep CNN due to their huge numbers of model parameters.

The proposed network architecture comprised only three convolutional layers and two fully connected layers with a small number of neurons. The choice of a smaller network design was motivated by the desire to reduce the risk of max-pooling as well as the nature of face recognition. All three colour channels were processed directly by the network. Images were first rescaled to 256-256 and a crop of 227-227 was fed to the network. The following three convolutional layers were defined:

- a) 96 filters of size 3-7-7 pixels are applied to the input in the first convolutional layer, followed by a rectified linear operator (ReLU), a max-pooling layer taking the maximal value of 3- 3 regions with two-pixel strides and a local response normalization layer.
- b) The 96 - 28- 28 output of the previous layer is then processed by the second convolutional layer, containing 256 filters of size 96- 5- 5 pixels. Again, this was followed by ReLU, a max-pooling layer, and a local response normalization layer with the same hyper parameters as before.
- c) Finally, the third and last convolutional layer operates on the 256 - 14- 14 blobs by applying a set of 384 filters of size 256 - 3 - 3 pixels, followed by ReLU and a max-pooling layer.

The following fully connected layers were then defined by:

- a) A first fully connected layer that receives the output of the third convolutional layer and contains 512 neurons, followed by a ReLU and a dropout layer.
- b) A second fully connected layer receives the 512-dimensional output of the first fully connected layer and again contains 512 neurons, followed by a ReLU and a dropout layer.
- c) A third, fully connected layer maps to the final classes for age or gender.

d) Finally, the output of the last fully connected layer was fed to a softmax layer that assigns a probability for each class. The prediction was made by selecting the class with the highest probability for the test image in question.

The research provided a shallow CNN with reduced parameters to prevent over-fitting. The size of the training data was inflated by artificially adding cropped versions of images in the training set. Two important contributions were made namely CNN can be used to provide improved age and gender classification results and the simplicity of the model implies that more elaborate systems may be capable of improving results. Since CNN was a deep learning algorithm, the time taken to train the networks depending on the dataset of images was long.

CNN Training for Face Photo-based Gender and Age Group Prediction with Camera was proposed in [24]. It is difficult to say that CNN trained with RGB colour images always produced good results in an environment where testing was performed with the camera rather than with image files. Therefore, with experiments, it has been observed that in camera-based testing CNN trained with grayscale images showed better gender and age group prediction accuracy than CNN trained with RGB colour images. The goal of the study was to compare CNN in two settings: file-based testing and camera-based testing in terms of gender and age group prediction accuracy. A brief description of the architecture of Google net with an explanation of the Adience data set used in experiments, along with CNN training options. Google net was employed for the gender and age group prediction experiments since it showed fairly good ImageNet objection classification accuracy and required relatively shorter training time. The main component of Google net was the inception block composed of  $1 \times 1$ ,  $3 \times 3$  and  $5 \times 5$  convolution layers. It consists of the first convolution layers of  $7 \times 7$  Convolution layer,  $3 \times 3$  max-pooling layer,  $1 \times 1$  Convolution layer,  $3 \times 3$  Convolution layer and final  $3 \times 3$  max-pooling layer. It was followed by nine consecutive inception blocks. A max-pooling layer came after the 2<sup>nd</sup> and the 7<sup>th</sup> inception block. A global average pooling layer came after the last inception block. To build a classifier, a fully connected layer is added after the global average pooling layer.

In both training and test data, each photo has two labels namely a gender label and an age group label. Some photos in the Audience dataset do not have gender labels or age labels. Only photos with two labels were included. Each photo in training data and test data was aligned so that the nose was located at the center of the photo to help CNN training and testing. Unlike the existing gender or age group classifiers, the CNN used was not initialized with pre-trained weights but with random uniform weights. In order to simplify the experiment, five-fold cross-validation usually used to get an averaged accuracy was not utilized. The focus was not on comparing the results with those of existing approaches, but on finding out what kind of image files or training methods were desirable for effective testing with the camera. It was observed that random brightness and contrast variations applied to RGB or

grayscale images were very effective in improving gender and age group prediction accuracy in camera-based testing.

Neural Network-based Age and Gender Classification for Facial Images was proposed in [25]. With huge volumes of training data, automatic face identification and verification from facial photos achieves acceptable accuracy, but face attribute recognition from facial photos remains difficult. As a result, developing an efficient and accurate face image categorization system based on face features is critical. The objective of the paper was to use biometric feature variation of males and females for classification and to classify images in different lighting conditions and different illumination conditions. The main goal of the proposed algorithm is to identify the corresponding age range and gender from human face images using a specific set of facial features.

The methodology proposed in this paper for automatic age and gender classification based on feature extraction from facial images makes use of biometric feature variation and two types of features namely, primary and secondary features and three main iterations namely preprocessing, feature extraction and classification. Classification was done using Artificial Neural Network (ANN). The main steps of the proposed age and gender classifier includes:

**a) Input image:** The input image was the source image intended to test with the age and gender classifier. User can input any type of image format like .jpg, .png, .tiff, and .bmp. The system will not accept face images with spectacles and images of children less than eight years of age.

**b) Face area detection:** The first phase will proceed to check whether the given input image contains a face part or not. The algorithm will reject the input image if there were no face area in the input image. Once the face area is detected, the classifier will extract the area and create a separate image per every face in the input image by cropping the background. Detected face images were preprocessed to standardize the face images by converting them to a unique format.

**c) Preprocessing:** Images used in the experiment were in different conditions such as the presence of noisy data, different lighting conditions, and different intensity levels. Thus, detected face images need to undergo a preprocessing step before forwarding to the classification stage, here, the detected face image was resized and converted to a grayscale image, also noise such as dirt on camera lenses, imperfections in-camera flash lighting was reduced.

**d) Feature extraction:** Relevant features important for the classification were extracted from the face images. Primary features such as eyes, nose, mouth, and eyebrow area and secondary features such as cheekbone, forehead, and eyelid area were extracted.

**e) Classification:** Classification was done in two main

steps. First, the gender classifier will identify the corresponding gender of the query image. After that, the image is transferred to the age classifier to identify the corresponding age group. Gender classification is done using shape variations of the features on the face and the age classification was based on the texture variations in the wrinkle areas.

The results obtained were:

Performance Evaluation of Gender Classification:

Test Set: Male - 85%, Female – 86.66%

Training Set: Male – 88.42%, Female – 91.42%

Performance Evaluation of Age Classification:

Test Set: (8-13) - 80%, (14-25) – 71.60%, (26-45) – 66.66%, (46-60) – 79.31%

Training Set: (8-13) – 71.11%, (14-25) – 82%, (26-45) – 67.27%, (46-60) – 96%

The contribution to knowledge was that the study proposed a methodology to classify the age and gender of facial images in the presence of different lighting and different illumination conditions. The limitation was that it was very difficult to find a facial image database with the corresponding age and gender data. Most of the images did not meet the constraints of the methodology. Therefore, collecting a large set of images from males and females from the four age categories according to the given lightning conditions, rotation conditions and facial expression was identified as a very challenging task in the research.

Automatic Age and Gender Classification using Supervised Appearance Model was proposed in [26]. The goal of the study was to provide an overview of the state of machine learning algorithms as applied to medical imaging, with an emphasis that will be most useful to the doctors and clinicians and provide impact in medical science and the patient healthcare system. The proposed Supervised Appearance Model (SAM) captures both shape and texture variability from the training dataset. This was done by forming a parameterized model using PLS dimensionality reduction to capture the variations as well as combine them in a single model. To build the supervised texture model, all face images were affinely warped to the mean shapes. This was done so that the control points of the training images match that of a fixed shape. Illumination variations are then normalized by applying a scaling and an offset to the warped images. The contributions to knowledge was the use of partial least-squares (PLS) regression in place of PCA. PLS is a dimensionality reduction technique that maximizes the covariance between the predictor and the response variable, thereby generating latent scores having both reduced dimension and superior predictive power. The model was termed the supervised appearance model (SAM). The feature extraction model is then applied to the problems of age estimation and

gender classification. The performance of the classifications is evaluated using the FGNET-AD benchmark database.

### Design of Multi-Cast Learning for Age and Gender Prediction

The performance of gender and age classification depends on face detection, facial image preprocessing, feature extraction and classification. The collection of a large, labelled image training set for this classification problem from social image repositories requires either access to personal information on the subjects on the images, usually their date of birth and gender, which is most often private, tasking or time consuming to manually label. Therefore, datasets for age and gender classification from real-world social images are limited. In this research, Audience and IDMB-Wiki Datasets were used. In this Section, the system overview, system architecture, system methodology and the software tools used in the project are discussed. The objective of the system is to implement both the CCN and Symbols for the recognition of the gender and estimation of the age of human beings in images. The comparative analysis of the results obtained and implementing a web interface for the evaluation of the models are reported.

### Architecture of the proposed system

The web-based evaluation system architecture as shown in Figure 8 below consists of a software known as Streamlit, an open-source library that makes it easy to create and share beautiful, custom web applications for machine learning and data science. It was built to bridge the gap between frontend and backend so that machine learning models can be deployed in hours instead of days. It takes care of both the design of the application and the API web server. The architecture also consists of the client, client storage and the camera. A client/user uses a camera/capturing device to take an image that the client wants the model to process, the image is stored in the client's device storage which is where all captured images of human faces are kept. The client/user sends a request through a browser to the web server from the client storage which was taken with a camera. Then python API web server receives the image as a request from the client and sends it to the machine learning model for processing. The model is the implementation of the age and gender prediction system which was trained with labelled data.

The data flow diagram of the age and gender prediction system of both the CNN and SVM models were illustrated below in Figure 9. It consists of the face dataset, data exploration phase, data preprocessing phase, feature extraction phase, data augmentation phase, apply algorithms phase, build model phase, compile model phase, train model phase, evaluate model phase and make predictions phase. In the development of the system, the tools that were used include:

- a) Scikit Learn Python Machine Learning Library
- b) Jupyter Notebook
- c) Keras Python Deep Learning Library
- d) Streamlit Python Library

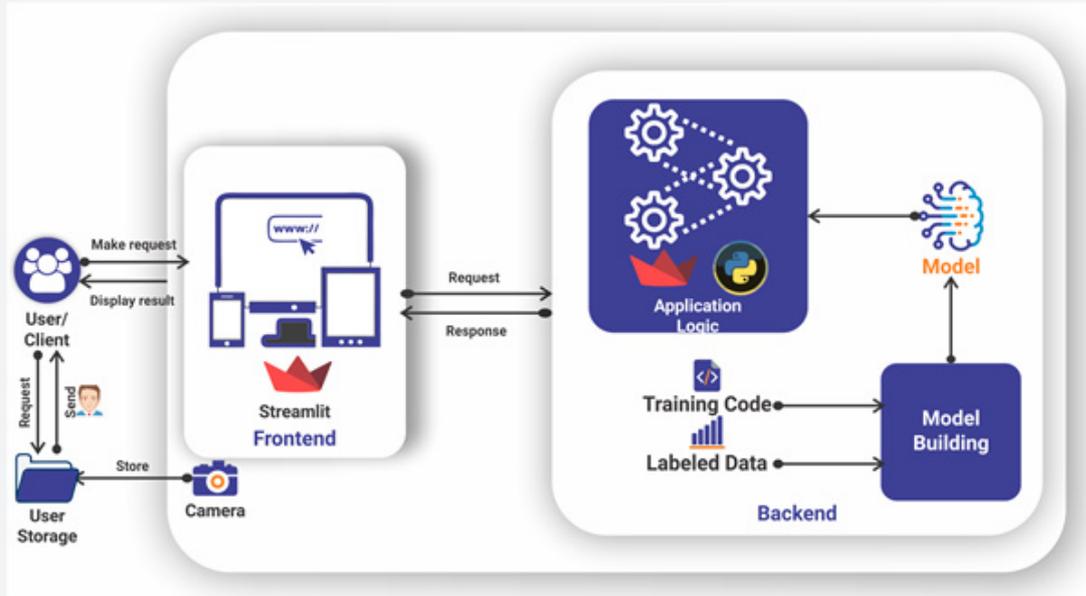


Figure 8: Architecture of proposed system.

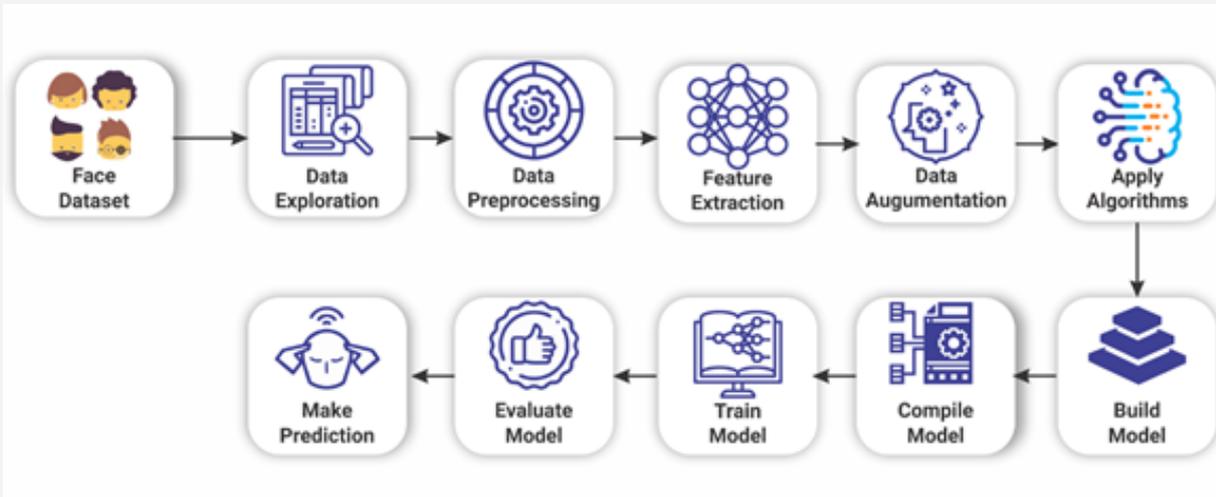


Figure 9: Data flow diagram of the prediction system.

**Scikit learn python machine learning library:** Scikit Learn christened Sklearn is a free machine learning library for Python. It is featured in various algorithms like SVM, Random Forests and K-Neighbours. It supports Python numerical and scientific libraries like Numpy and SciPy. It provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. The library is built upon the SciPy (Scientific Python)

that must be installed before Sklearn is used and it contains a lot of efficient tools for machine learning and statistical modelling including classification, regression, clustering, and dimensionality reduction. The stack includes:

- a) Numpy: Base n-dimensional array package
- b) SciPy: Fundamental library for scientific computing

- c) Matplotlib: Comprehensive 2D/3D plotting
- d) IPython: Enhanced interactive console
- e) Sympy: Symbolic mathematics
- f) Pandas: Data structures and analysis.

**Jupyter notebook:** Jupyter Notebook is an open-source web application that allows for the creation and sharing of documents that contain live code, equations, visualizations and narrative texts. It is used for data cleaning and transformation, numerical simulation, statistical modelling, data visualization, machine learning and so on. It is an interactive web tool known as a computational notebook, which researchers can use to combine software code, computational output, explanatory text and multimedia resources in a single document.

**Keras python deep learning library:** Keras is an open-source neural network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, R, Theano or PlaidML. Designed to enable fast experimentation with deep neural networks, it is user-friendly, modular and extensible. It is an API designed for human beings, not machines, following best practices for reducing cognitive load, it offers consistent and simple APIs. It minimizes the number of user actions required for common use cases and it provides clear and actionable error messages. It also has extensive documentation and developer guides. Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural networks code.

**Streamlit python library:** Streamlit is a free open source Python library that makes it easy to create and share beautiful, custom web applications for machine learning and data science. It is a data application framework that makes it easy to build and deploy powerful data applications in just a few minutes instead of days. It was launched in 2018 to help scientists build data applications emphasizing fast development with pure Python approach. It is easy to learn and use and the API is straightforward.

### Methodology of proposed research

The methodology used in the study includes data preparation, data pre-processing, feature extraction and classification and prediction.

**Data preparation:** In this study, datasets are needed for training, validating and testing the models. The organization of the dataset is also very essential since each of them must follow the same pattern of age prediction and gender classification into male and female. Two datasets were used in this research:

**UTKFace dataset:** It is a large-scale face dataset with a long age span (ranging from 0 to 116 years old). The dataset consists

of over 20,000 face images with annotations of age, gender and ethnicity. The images cover large variations in pose, facial expression, illumination, occlusion, resolution [27].

**Adience dataset:** It is composed of pictures taken by a camera from smartphones or tablets. The images of the dataset capture extreme variations, including extreme blur (low-resolutions), occlusions, out-of-plane pose variations, expressions. The dataset consists of 26,500 images distributed in eight age categories [(0 - 2), (4 - 6), (8 - 12), (15 - 20), (25 - 32), (38 - 43), (48 - 53), (60 - 100)].

**Data preprocessing:** In this stage, brightness and contrast are normalized, the face image geometric features are improved and the image size (the number of pixels) is reduced. Pre-processing includes three steps: resizing the image, converting to grayscale and noise-reduced image. The input image is resized and the color images are converted to grayscale. Data augmentation is also applied to the images in this stage, where each image is flipped, mirrored and rotated to generate more training datasets and to avoid overfitting [28-35]. The feature extraction is carried out in two ways. First, there is the CNN model which can be thought of as a combination of two components, namely: feature extraction and classification. The convolution and the pooling layers perform feature extraction, given an image, the convolution layer detects features such as two eyes, two ears, a nose, long hair, cheekbone, and so on. The fully connected layer then acts as a classifier on top of these features and assigns a probability that the input image exists as a male and an age group. Second, the SVM model which feature extraction and selection that are very important in this model because reducing the number of features in machine learning plays a major role especially when working with large datasets which can speed up training, avoid overfitting and ultimately lead to better classification results. Feature importance can therefore be determined by comparing the size of the classifier coefficients to each other. By looking at the SVM coefficients, it is possible to identify the main features used in classification and get rid of the ones that hold less variance [36-45].

**Model implementation:** The CNN model and SVM models are implemented to perform age estimation and gender classification on input images. The gender has two classes, namely: male and female and age has eight classes namely: (0 - 2), (4 - 6), (8 - 12), (15 - 20), (25 - 32), (38 - 43), (48 - 53) and (60 - 100).

**a) CNN model:** The CNN model to be implemented contains four convolutional layers, each followed by a rectified linear operation (a mathematical function that is added to a dense layer, which gives it more power to solve complex problems or non-linear problems) and a pooling layer (MaxPooling which is used with a  $2 \times 2$  stride, that reduces the size of the image). The model begins by instantiating the sequential model which takes an array of Keras layers (Convolutional, Max Pooling, and Softmax) before training begins. After compilation, it is trained by specifying the

number of epochs (iterations over the entire dataset) to train for. The model is saved so that it can be reloaded whenever it is needed.

**b) SVM model:** SVM takes the training datasets and finds a hyperplane between the eight classes, for the age namely: (0 - 2), (4 - 6), (8 - 12), (15 - 20), (25 - 32), (38 - 43), (48 - 53), (60 - 100) and two classes for gender namely: male and female. It tries to make a decision boundary in such a way that the separation between the classes is as wide as possible and then it predicts the accurate age and gender-based on its result. To train the algorithm, the fit method of the SVC class of Sklearn is called and the training data is passed as a parameter to this method [46-48].

### Implementation of the Proposed System

In this section, the code implementation of the CNN model and models SVM model that were used for analysis and prediction are considered. The implementation of the code for the web interface and the connecting Application Programming Interface (API) for the models was also considered and discussed. Lastly, the results obtained from using the two models are presented using the web interface and an attempt was made to analyze and compare both

results. The minimum hardware requirements of the system are 32 bit, 64-bit Intel and AMD processors, 4gb RAM and 128Gb Hard Drive. The minimum software requirements of the system were any Operating System and Web browser.

### Convolutional Neural Network (CNN) – Model I

**a) Importation of libraries:** All the essential libraries that were used in the research were imported so that it could be available for use in Python. Figure 10 presents the libraries namely: Numpy, Pandas, Matplotlib, Keras, Sklearn and some inbuilt libraries such as math, os and glob. The Numpy library is a general-purpose array processing package that provides a high-performance multidimensional array object and tools for working with the arrays. It is essentially the fundamental package for scientific computing with Python. The Pandas library is primarily written for data manipulation and analysis, offering data structures and operations for manipulating numerical tables and time series. Matplotlib is a multi-platform data visualization library used for 2D plots of arrays. The inbuilt module provides functions for interacting with the operating system, math module provides access to mathematical functions while the glob module is used to retrieve files or pathnames matching a specific pattern.



```
In [1]: import numpy as np
import pandas as pd
import os
import glob
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Figure 10: Library importation of the models.

**b) Initializing constants:** After the necessary libraries were imported, some constants such as the dataset folder name, the fraction of the dataset to be used for testing and the image size of the model were initialized as shown in Figure 11.

**c) Extraction of age and gender from image:** Every image in the dataset contains information such as the age and gender in the name of the file. The names of files appeared like this: “17\_1\_0\_20170109214048004.jpg.chip.jpg” where the first number before the underscore represents the age, the second number represents the gender namely: male = 0 and female = 1, the third number represents the race of the person in the image but this information is not needed in this research. This step also involves dropping empty columns using the .dropna() method.

The code for generating age group and gender is presented in Figure 12.

**d) Visualizing the dataset:** Using plotly library, a pie-chart is created to visualize the dataset, both gender and age distribution, illustrated in the Figure 13 & Figure 14 below.

**e) Data preprocessing:** After the dataset has been visualized and empty columns dropped, the dataset is split into train and test images using the ratio 70:30, the train dataset is further splitted into training and validation dataset. The “get\_data\_generator” function was implemented which makes use of python’s Image library to convert each of the images to an array of image features. The code for carrying out these is presented in Figure 15.

```
In [2]: DATASET_FOLDER_NAME = './data/UTKFace'

#fraction of the dataset to be used for testing
TRAIN_TEST_SPLIT = 0.7

# image size for model
IM_WIDTH = IM_HEIGHT = 198

In [3]: ID_GENDER_MAP = {0:'male', 1:'female'}
        GENDER_ID_MAP = {'male': 0, 'female': 1}
```

Figure 11: Initializing constants for the CNN model.

```
In [4]: def parse_dataset(dataset_path, ext='jpg'):
        """
        Used to extract information about our dataset. It does iterate over all images and return a DataFrame with
        the data (age and gender) of all files.
        """
        def parse_info_from_file(path):
            """
            Parse information from a single file
            """
            try:
                filepath, filename = os.path.split(file)
                age, gender, _ = filename.split("_")
                return age, ID_GENDER_MAP[int(gender)]
            except Exception as e:
                print(e)
                return None, None

        files = glob.glob(os.path.join(dataset_path, "%.*s" % ext))

        records = []
        for file in files:
            info = parse_info_from_file(file)
            records.append(info)

        df = pd.DataFrame(records)
        df['file'] = files
        df.columns = ['age', 'gender', 'file']
        df = df.dropna()

        return df
```

```
In [7]: df = parse_dataset(DATASET_FOLDER_NAME)
        df.head()

not enough values to unpack (expected 4, got 3)
not enough values to unpack (expected 4, got 3)
not enough values to unpack (expected 4, got 3)
```

```
Out[7]:
```

	age	gender	file
0	100	male	./data/UTKFace/100_0_0_20170112213500903.jpg.c...
1	100	male	./data/UTKFace/100_0_0_20170112215240346.jpg.c...
2	100	female	./data/UTKFace/100_1_0_20170110183726390.jpg.c...
3	100	female	./data/UTKFace/100_1_0_20170112213001988.jpg.c...
4	100	female	./data/UTKFace/100_1_0_20170112213303693.jpg.c...

Figure 12: Extracting age and gender from the image.

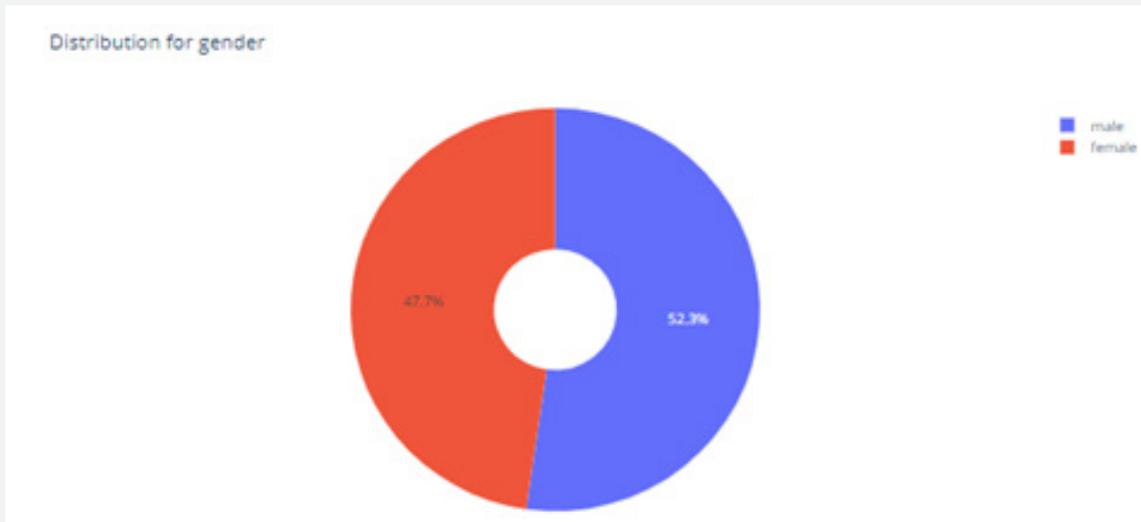


Figure 13: Visualizing the gender dataset.

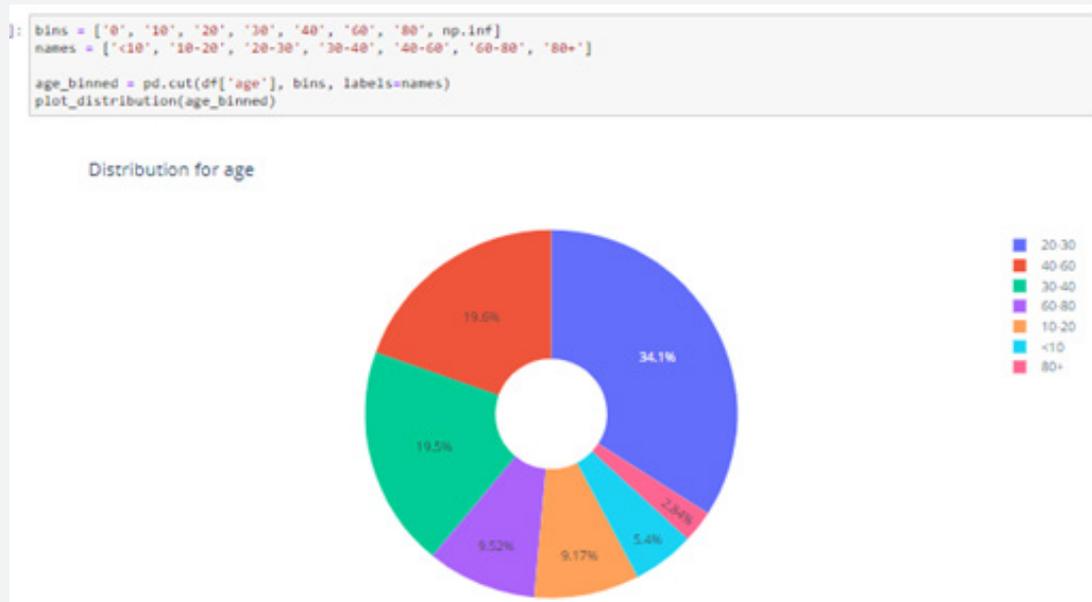


Figure 14: Visualizing the age group dataset.

f) **CNN model implementation:** The model consists of six convolutional layers of filters 16, 32, 64, 128, 256 and 256 respectively each separated by a normalization Layer and a MaxPooling Layer as presented in Figure 16.

**Support vector machine – model II**

a) Loading the dataset in SVM model is presented in Figure 17.

Visualizing the dataset and splitting into test and train dataset with ratio 70:30 is shown in Figure 18.

b) Building the SVM Model is presented in Figure 19.

f 17 18 19

The comparative analysis of CNN and SVM is presented in Table 3.

**Table 3:** Comparative Analysis of CNN and SVM.

Models	Gender Accuracy	Gender Error Rate	Age Mean Absolute Error	Age Accuracy
Convolutional Neural Network	90.14%	9.80%	60.33%	39.67%
Support Vector Machine	80.67%	19.33%	59.89%	40.11%

```

In [14]: p = np.random.permutation(len(df))
train_up_to = int(len(df) * TRAIN_TEST_SPLIT)
train_idx = p[:train_up_to]
test_idx = p[train_up_to:]

# split train_idx further into training and validation set
train_up_to = int(train_up_to * 0.7)
train_idx, valid_idx = train_idx[:train_up_to], train_idx[train_up_to:]

df['gender_id'] = df['gender'].map(lambda gender: GENDER_ID_MAP.get(gender))
max_age = int(df['age'].max())

In [15]: from tensorflow.keras.utils import to_categorical
from PIL import Image

def get_data_generator(df, indices, for_training, batch_size=16):
    images, ages, genders = [], [], []
    while True:
        for i in indices:
            r = df.iloc[i]
            file, age, gender = r['file'], int(r['age']), r['gender_id']
            im = Image.open(file)
            im = im.resize((IM_HEIGHT, IM_WIDTH))
            im = np.array(im) / 255.0
            images.append(im)
            ages.append(age / int(max_age))
            genders.append(to_categorical(gender, 2))
            if len(images) >= batch_size:
                yield np.array(images), [np.array(ages), np.array(genders)]
                images, ages, genders = [], [], []
        if not for_training:
            break

```

**Figure 15:** Data preprocessing for the CNN model.

## Conclusion

In this paper, based on some datasets, the comparative analysis of CNN model and SVM model was carried out for both age estimation and gender classification. The CNN model is made up of six convolutional layers that are separated by batch normalization and maxpooling, whereas the SVM model was created using the sklearn SVC method. A web-based evaluation system was also developed which is capable of receiving image inputs, detecting faces and making use of the models to classify the gender and also estimate the age of the faces detected in the image. By extending the training image dataset and ensuring that the bias in the training image collection is eliminated and it captures more diverse human faces.

The approach utilized in this study can be significantly improved. In addition, further research could expand the application of this proposed approach to recognize deeper social characteristics of user identities, such as occupation and educational level. The proposed method could possibly be used to classify human expressions and diagnose facial diseases. The limitations in this study includes: Lack of computation resources: A sufficient number of computation resources is required to train enough dataset for the models to perform well, allowing the models to be trained faster on larger datasets. Inbalanced dataset: The images used to train and test the models do not represent the whole range of human facial variety, which could lead to bias in the models' results.

```

In [20]: from keras.layers import Input, Dense, BatchNormalization, Conv2D, MaxPool2D, GlobalMaxPool2D, Dropout
         from tensorflow.keras.optimizers import SGD
         from keras.models import Model

         def conv_block(inp, filters=32, bn=True, pool=True):
             _ = Conv2D(filters=filters, kernel_size=3, use_bias=False, activation='relu')(inp)
             if bn:
                 _ = BatchNormalization()(_)
             if pool:
                 _ = MaxPool2D()(_)
             return _

         input_layer = Input(shape=(IM_HEIGHT, IM_WIDTH, 3))
         _ = conv_block(input_layer, filters=16, bn=False, pool=False)
         _ = conv_block(_, filters=32)
         _ = conv_block(_, filters=64)
         _ = conv_block(_, filters=128)
         _ = conv_block(_, filters=256)
         _ = conv_block(_, filters=256)
         bottleneck = GlobalMaxPool2D()(_)

         # for age calculation
         _ = Dense(units=128, activation='relu')(bottleneck)
         age_output = Dense(units=1, activation='sigmoid', name='age_output')(_)

         # for gender prediction
         _ = Dense(units=128, activation='relu')(bottleneck)
         gender_output = Dense(units=len(GENDER_ID_MAP), activation='softmax', name='gender_output')(_)

         model = Model(inputs=input_layer, outputs=[age_output, gender_output])

In [21]: model.compile(optimizer='rmsprop',
                      loss={'age_output': 'mse', 'gender_output': 'categorical_crossentropy'},
                      loss_weights={'age_output': 2., 'gender_output': 1.},
                      metrics={'age_output': 'mae', 'gender_output': 'accuracy'})
         model.summary()

In [20]: from keras.layers import Input, Dense, BatchNormalization, Conv2D, MaxPool2D, GlobalMaxPool2D, Dropout
         from tensorflow.keras.optimizers import SGD
         from keras.models import Model

         def conv_block(inp, filters=32, bn=True, pool=True):
             _ = Conv2D(filters=filters, kernel_size=3, use_bias=False, activation='relu')(inp)
             if bn:
                 _ = BatchNormalization()(_)
             if pool:
                 _ = MaxPool2D()(_)
             return _

         input_layer = Input(shape=(IM_HEIGHT, IM_WIDTH, 3))
         _ = conv_block(input_layer, filters=16, bn=False, pool=False)
         _ = conv_block(_, filters=32)
         _ = conv_block(_, filters=64)
         _ = conv_block(_, filters=128)
         _ = conv_block(_, filters=256)
         _ = conv_block(_, filters=256)
         bottleneck = GlobalMaxPool2D()(_)

         # for age calculation
         _ = Dense(units=128, activation='relu')(bottleneck)
         age_output = Dense(units=1, activation='sigmoid', name='age_output')(_)

         # for gender prediction
         _ = Dense(units=128, activation='relu')(bottleneck)
         gender_output = Dense(units=len(GENDER_ID_MAP), activation='softmax', name='gender_output')(_)

         model = Model(inputs=input_layer, outputs=[age_output, gender_output])

In [21]: model.compile(optimizer='rmsprop',
                      loss={'age_output': 'mse', 'gender_output': 'categorical_crossentropy'},
                      loss_weights={'age_output': 2., 'gender_output': 1.},
                      metrics={'age_output': 'mae', 'gender_output': 'accuracy'})
         model.summary()

```

Figure 16: Model implementation for the convolutional neural network.

```

In [2]: def imshow(img):
        plt.imshow(img, interpolation="bicubic")
        plt.xticks([], plt.yticks([]))
        plt.show()

In [3]: onlyfiles = os.listdir("./data/utkface_aligned_cropped/UTKFace")
        y = np.array([[i.split('_')[0]],[i.split('_')[1]] for i in onlyfiles])
        print(y.shape)
        print(y[0])

        (23708, 2, 1)
        [['100']
         ['0']]

In [42]: IM_WIDTH = IM_HEIGHT = 50
        X_data = []
        for file in onlyfiles:
            face = cv2.imread('./data/utkface_aligned_cropped/UTKFace/' + file, cv2.IMREAD_COLOR)
            face = cv2.cvtColor(face, cv2.COLOR_BGR2RGB)
            face = cv2.resize(face, (IM_WIDTH,IM_HEIGHT))
            X_data.append(face)
        X_data = np.array(X_data)
        X_data.shape

Out[42]: (23708, 200, 200, 3)

```

Figure 17: Loading the dataset for the support vector machine.

```

In [5]: X = np.squeeze(X_data)
        imshow(X[1])
        print(y[1])

        [['100']
         ['0']]

In [34]: X_train, X_valid, y_train, y_valid = train_test_split(X, y, test_size = 0.3)
        y_train = [y_train[:,1], y_train[:,0]]
        y_valid = [y_valid[:,1], y_valid[:,0]]
        X_train.shape

Out[34]: (16595, 50, 50, 3)

```

Figure 18: Visualizing the dataset for the SVM model.

```
[41]: from sklearn.svm import SVC
      from sklearn.ensemble import RandomForestClassifier
      from sklearn.metrics import accuracy_score

      # Training on dataset
      rf_clf = RandomForestClassifier(random_state=42)
      rf_clf.fit(X_train, y_train)

      model = SVC(C=1, kernel='poly', gamma='auto')
      model.fit(X_train, y_train)
```

Figure 19: Model implementation of support vector machine.

## References

- Maina BA, Ugail H, Connah D (2016) Automatic Age and Gender Classification using Supervised Appearance Model. Journal of Electronic Imaging 25(6).
- Syantani G, Kumar S (2015) Gender Classification and Age Detection based on Human Facial Features using Multi-Class SVM. British Journal of Applied Science and Technology 10(4): 1-15.
- Dileep MR, Danti A (2018) Human Age and Gender Prediction Based on Neural Networks and Three Sigma Control Limits. Article in Applied Artificial Intelligence 32(3).
- Zhang K, Liu N, Yuan X, Guo X, Gao Ce, et al. (2019) Fine-Grained Age Estimation in the Wild with Attention LSTM Networks. IEEE Transactions on Circuits and Systems for Video Technology 30(9): 3140-3152.
- Mandy S (2017) Introduction to Types of Classification Algorithms in Machine Learning.
- Andrea M (2011) Credit Card Detection using k-star Machine Learning. Third Biennial Conference on Transition from Observation to Knowledge to Intelligence.
- Savory SE (1998) Artificial Intelligence and Expert Systems. Ellis Horwood Publishers. Halsted Press, John Wesley and Sons. New York.
- Liu B, Li X, Lee WS, Yu PS (2004) Text classification by labeling words. Proceedings of the National Conference on Natural Language Processing & Information Extraction USA 42: 54-69.
- Ko Y, Park J, Seo J, Choi S (2007) Using Classification Techniques for Informal. Requirements in the Requirements Analysis-Supporting System. Elsevier Journal of Information and Software Technology 49(11-12): 1128-1140.
- Adegoke MA (2017) Data Driven Latent Dirichlet Allocation Model. PhD Thesis in the School of Postgraduate Studies University of Lagos, Nigeria.
- Adegoke MA, Ayeni JOA, Adewole PA (2019) Empirical Prior Latent Dirichlet Allocation Model. Nigerian Journal of Technology (NIJOTECH) 38(1): 223-232.
- Salton G, Yang CS, Yu CT (1975) A Theory of Term Importance in Automatic Text Indexing. Communications of the ACM 18: 613-620.
- Deerwester S, Dumais ST, Frnas GW, Landauer TR, Harshman R (1990) Indexing by Latent Semantic Analysis. Journal of the American Society of Information Science 41(6): 391-407.
- Hofmann T (2001) Unsupervised Learning by Probabilistic Latent Semantic Analysis. Machine Learning Journal 42: 177-196.
- Blei DM (2012) Probabilistic Topic Models. Communications of the ACM 55(4): 77-84.
- Prabhu M (2018) Understanding of Convolutional Neural Network (CNN) - Deep Learning.
- Sumit S (2018) A Comprehensive Guide to Convolutional Neural Network - the ELI5 way.
- Eidinger E, Enbar R, Hassner T (2013) Age and Gender Estimation of Unfiltered Faces. IEEE Transactions on Information Forensics and Security 9(12): 2170-2179.
- Melange PA, Sable GS (2018) Age Group Estimation and Gender Recognition using Face Feature. The International Journal of Engineering and Science (IJES), 7(7): : 23-19.
- Nurul AT, Kumar ED (2018) A Gender Recognition System from Facial Image. International Journal of Computer Applications 180(23): 5-14.
- Ekmekji A (2016) Convolutional Neural Networks for Age and Gender Classification. Stanford University.
- Ulman (2020) Artificial Intelligence in Industry. Ulma Advanced Forged Solution. New Jersey.
- Levi G, Hassner T (2015) Age and Gender Classification, using Convolutional Neural Networks. Department of Mathematics and Computer Science, the Open University of Israel.
- Jhang K, Cho J (2019) CNN Training for Face Photo-based Gender and Age group Prediction with Camera. Conference Paper, pp. 548-551.
- Thakshila RK, Anuja TD (2015) Neural Network Based Age and Gender Classification for Facial Images. International Journal on Advances in ICT for Emerging Regions 7(2): 1-10.
- Bukar AM, Ugail H, Connah D (2016) Automatic Age and Gender Classification using Supervised Appearance Model. The University of Bradford, Faculty of Engineering and Informatics 25.

27. UTK Face Dataset (2020) The challenges of using Machine Learning to identify Gender in Images by Stefan Wofjcik and Emma Remy.
28. Kwon YH, Lobo NV (1999) Age Classification from Facial Images. *Computer Vision and Image Understanding* 74(1): 1-21.
29. Bing HBW, Ping LC, Wen CC (2001) Classification of Age Groups Based on Facial Features. *Tamkang Journal of Science and Engineering* 4(3): 183-192.
30. Stawska Z, Milczarski P (2013) Gender Recognition Methods Useful in Mobile Authentication Applications. Department of Computer Science, Faculty of Physics and Applied Informatics, University of Lodz, Poland 5(2): 248-259.
31. Adegoke MA, Ayeni JOA (2014) Design of Analysis Supporting Tool for Large Scale Systems Based on Probabilistic Latent Semantic Indexing. On the Dynamics of Evolving Knowledge Society, University of Lagos Festchrift Group pp. 149-160.
32. Martins, Berry (2007) *Handbook of Latent Semantic Analysis*. Erlbaum Publishers.
33. Minsky M (1982) Steps Towards Artificial Intelligence. *Proceedings of IRE* 49(1): 8-30.
34. Turney D, Pantel P (2010) From Frequency to Meaning: Vector Space Models of Semantics. *Journal of Artificial Intelligence Research, Canada* 37(1): 141-188.
35. Olszewska JI (2016) Automated Face Recognition: Challenges and Solutions.
36. Lee JH, Chan YM (2018) Joint Estimation of Age and Gender from Unconstrained Face Images using Lightweight Multi-task CNN for Mobile Applications.
37. Ngugen DT, Cho SR, Shin KY, Bang JW, Park KR (2018) Comparative Study of Human Age Estimation with or without Pre-classification of Gender and Facial Expression.
38. Rastogi A, Aneja M (2018) Gender Classification from Facial Images; Performing the classification of Gender from the facial images and implementing the same on real-time. Bharati Vidyapeeth's College of Engineering, New Delhi, India.
39. Tawhid NA, Dey EK (2018) A Gender Recognition System from Facial Image. *International Journal of Computer Applications* 18(3): 5-14.
40. Khan K, Attique M (2019) Automatic Gender Classification through Face Segmentation.
41. Rodriguez P, Cucurull G, Gonfaus JM, Roca FX, Gonzalez J (2019) Age and Gender Recognition in the Wild with Deep Attention. *Computer Vision Center and University Autònoma de Barcelona (UAB)*, 08193 Bellaterra, Catalonia Spain 72: 563-571.
42. Savchenko AV (2019) Efficient facial representations for age gender and identity recognition in organizing photo albums using multi-output Conv Net. *PeerJ Comput Sci*.
43. IMDB Wiki Faces Dataset. Benchmark Gender and Age Classification.
44. (2020) Gender and Age Classification using OpenCV Deep Learning (C++/Python) by Vikas Gupta.
45. Rushikesh P (2018) Support Vector Machines (SVM) An Overview.
46. Wikipedia (2020) Support-Vector Machine.
47. Wikipedia (2020) Statistical Classification.
48. (2020) Difference between CNN and SVM.



This work is licensed under Creative Commons Attribution 4.0 License  
DOI: [10.19080/TTSR.2022.05.555667](https://doi.org/10.19080/TTSR.2022.05.555667)

### Your next submission with Juniper Publishers will reach you the below assets

- Quality Editorial service
- Swift Peer Review
- Reprints availability
- E-prints Service
- Manuscript Podcast for convenient understanding
- Global attainment for your research
- Manuscript accessibility in different formats  
( Pdf, E-pub, Full Text, Audio)
- Unceasing customer service

Track the below URL for one-step submission  
<https://juniperpublishers.com/online-submission.php>