sustainability

Article

Convolutional Neural Network-Based Personalized Program Recommendation System for Smart Television Users

Khasim Vali Dudekula [1,*], Hussain Syed [1,*], Mohamed Iqbal Mahaboob Basha [1], Sudhakar Ilango Swamykan [1], Purna Prakash Kasaraneni [1], Yellapragada Venkata Pavan Kumar [2], Aymen Flah [3] and Ahmad Taher Azar [4,5]

1 School of Computer Science and Engineering, VIT-AP University, Amaravati 522237, Andhra Pradesh, India
2 School of Electronics Engineering, VIT-AP University, Amaravati 522237, Andhra Pradesh, India
3 Energy Processes Environment and Electrical Systems Unit, National Engineering School of Gabes, University of Gabes, Gabès 6072, Tunisia
4 College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia
5 Faculty of Computers and Artificial Intelligence, Benha University, Benha 13518, Egypt
* Correspondence: d.khasimvali@vitap.ac.in (K.V.D.); hussain.syed@vitap.ac.in (H.S.)

Abstract: The smart home culture is rapidly increasing across the globe and driving smart home users toward utilizing smart appliances. Smart television (TV) is one such appliance that is embedded with smart technology. The users of smart TV have their interests in the programs. However, automatic recommendation of programs for user-to-user is still under-researched. Several papers discussed recommendation systems, but those are related to different applications. Even though there are some works on recommending programs to smart TV users (single-user and multi-user), they did not discuss the smart TV camera module to capture and validate the user image for recommending personalized programs. Hence, this paper proposes a convolutional neural network (CNN)-based personalized program recommendation system for smart TV users. To implement this proposed approach, the CNN algorithm is trained on the datasets 'CelebFaces Attribute Dataset' and 'Labeled Faces in the Wild-People' for feature extraction and to detect a human face. The trained CNN model is applied to the user image captured by using the smart TV camera module. Further, the captured image is matched with the user image in the 'synthetic dataset'. Based on this matching, the hybrid filtering technique is proposed and applied; thereby the recommendation of the respective program is done. The proposed CNN algorithm has achieved approximately 95% training performance. Besides, the performance of hybrid filtering is approximately 85% from the single-user perspective and approximately 81% from the multi-user perspective. From this, it is observed that hybrid filtering outperformed conventional content-based filtering and collaborative filtering techniques.

Keywords: artificial intelligence (AI); convolutional neural network (CNN); hybrid filtering; machine learning; program recommendation system; smart appliances; home automation; smart Television (TV)

Citation: Dudekula, K.V.; Syed, H.; Basha, M.I.M.; Swamykan, S.I.; Kasaraneni, P.P.; Kumar, Y.V.P.; Flah, A.; Azar, A.T. Convolutional Neural Network-Based Personalized Program Recommendation System for Smart Television Users. *Sustainability* **2023**, *15*, 2206. https://doi.org/10.3390/su15032206

Academic Editor: Muhammad Syafrudin

Received: 21 November 2022
Revised: 14 January 2023
Accepted: 15 January 2023
Published: 25 January 2023

1. Introduction

A recommendation system is also known as a recommender system, which filters the information and accordingly predicts the rating or priority a customer gives to an item. These systems are predominantly used in business applications. They enable the user to experience a better product search by receiving personalized program recommendations. Presently, the smart home culture is increasing day by day [1–4], where all appliances are expected to work smartly to continuously meet user requirements. Besides, by operating smartly, these are expected to save electrical energy. Embedding smart technology [5] in a normal television (TV) transforms it into a smart TV, and it has become one of the smart appliances in the smart home. Further, the apps in the smart TV are providing a comfortable, flexible, and controlled program experience to the users. There is a necessity of testing the usability of these apps through an automated model [6]. In general, smart

TV programs are shared by all family members, but a personalized program interface based on user preferences is required [7]. The reason for this requirement is that the family members will have their preferences in watching programs. This is the prime focus of the research to develop a personalized program recommendation system for a single or group of members based on their preferences. The recent technological growth has enabled artificial intelligence (AI)-based recommendation systems in playing a key role in several applications [8–10]. Additionally, various machine learning algorithms have been implemented for recommender systems in the field of AI [11]. The gaining importance for the recommendation systems driving several applications towards implementing them as discussed in Section 1.1.

1.1. Related Work

Recommendation systems are not only limited to content and collaborative filtering, but also are extended to contextual situations. In this regard, an extensive survey was conducted on the computational intelligence methods such as genetic algorithm, particle swarm optimization, artificial neural networks, k-nearest neighbor, Bayesian theory, support vector machine, fuzzy sets, etc. for improving conventional design using contextual situations [12–20]. Further, a situation-aware recommender system based on the recurrent neural network was developed for personalized healthcare applications [21]. A personality-aware recommendation system was discussed in [22] for personality computing based on the models of personality types and traits. A personalized channel recommendation system based on the personal preferences of the viewers was implemented in [23] for live streaming platforms. This was implemented using user preference clustering and hybrid user preference clustering. Additionally, a widespread survey was conducted on the technologies i.e., association rules, matrix factorization, hybrid recommendation, etc. for personalized news recommendation systems [24]. A framework that consists of incremental feature scanning with multiple windows and a hierarchical behavior structure was implemented to associate the multiple modes of the transport system with the recommendation system [25]. A recommender system using a support vector machine was developed to facilitate the tourists by providing support to their decision-making [26]. An item-to-item recommender system based on k-means clustering was discussed in [27] to create image mosaicking. A nutrient recommendation system based on an improved genetic algorithm was discussed to suggest the appropriate nutrients based on the production and fertility of the crop [28]. A hierarchical recommendation system using item description and the review-based deep sequential recommendation was presented in e-commerce applications to recommend products to customers based on the given reviews [29]. A home energy recommendation system based on deep reinforcement learning was developed based on user activities and feedback [30]. A demand-side personalized recommendation system using a non-intrusive appliance load monitoring technique was discussed in [31] to save the appliances' energy in the residencies connected to the smart grid. A thorough survey was conducted on the features and challenges of recommendation systems while integrating them with blockchain technology [32]. All the abovementioned recommendation systems were focused on the likes of users and the recommendations functioned accordingly. However, it is also necessary to focus on how to predict and prepare the users' dislikes list. To predict this list, an approach named signed network-based inference was discussed in [33]. Along with this, it is also important to know the quality of recommendations, which reveals the working condition of the recommendation system. In this aspect, a survey was conducted on the metrics and strategies that are useful for the evaluation of recommendations further to know the effectiveness of the recommendation system [34]. The abovementioned literature works discussed recommendation systems related to various applications and fields. In addition to the above, the literature works related to smart TV recommendation systems are discussed as follows.

A thorough review of various opportunities, challenges, and directions for future research in smart TV personalized content recommendation systems was conducted. Further, the limitations of various recommendation approaches i.e., content-based filtering, collaborative filtering, contextual-based, etc., were discussed in [35]. Novel formula and age-gender matrix methodology were implemented in [36] to generate user profiles for improving the outcome of the recommendation system by detecting the users' dominance in a group. A neural network-based viewing environment model "deepTV" was implemented in live TV recommendation systems to solve the issue of cold-start [37]. The combination of genetic algorithms, hybrid broadcast broadband TV, and artificial neural networks was implemented to record the users' TV viewing habits and further create user profiles. Based on these profiles, the recommendation of programs will be done [38]. A multiple-study was conducted on the smart TV user experience factors to learn the user satisfaction and the real-world challenges for TV manufacturers [39]. Further, [40] discussed the factors such as history, feedback, etc. that affect the recommendation systems' performance in a smart TV environment. A content-based recommendation method was implemented in [41] to calculate the content consumption calculation in smart TV program recommendations. A one-class classification technique was implemented to recommend personalized programs [42]. The unification of two latent Dirichlet allocation models was employed to recommend the programs to a similar group of TV users [43]. The novel research field successfully combined the machine learning and swarm intelligence approaches and proved to be able to obtain outstanding results in different areas [44,45].

1.2. Paper Contribution

From the above-described literature works, it is observed that several papers discussed different applications of recommendation systems in recommending programs in the smart TV environment. To the best of the authors' knowledge, there is no such work that presents the capturing and validation of user images through a smart TV camera module. Further, the recommendation of personalized programs is based on the priority of the users and their existing channel/program interests. With this motivation, this paper proposes a CNN-based personalized program recommendation system for smart TV users. The key contributions of the proposed work are summarized as follows:

- The CNN algorithm is trained on the datasets 'CelebFaces Attribute Dataset' and 'Labeled Faces in the Wild-People' for feature extraction and to detect a human face.
- The trained CNN model is applied to the smart TV user image that is captured by the smart TV camera module. Further, this captured image is matched with the smart TV user image that is already stored in the smart TV storage, i.e., 'synthetic dataset'. This facial information is stored in a local database along with the television and used for further processing. Nowadays, smart TVs have in-built memory and storage. For the proposed work, this storage is used for storing the facial information of respective family members. Hence, there is no need to store the facial images in a central repository. Since the facial information in terms of the derived features of the respective family members instead of human faces is stored in their smart TV storage itself, the possibility for security and ethical issues is very low.
- Based on this matching, the filtering techniques, namely content-based filtering, collaborative filtering, and hybrid filtering techniques are applied thereby to recommend the programs from single-user and multi-user perspectives. During the collaborative filtering process, the sparsity issue was faced in the feature vector. This issue was addressed by using the matrix factorization technique across the user profiles.
- Among these filtering techniques, the hybrid filtering technique outperformed in recommending the programs to both single-user and multi-user perspectives.

The remaining sections of the paper are organized as follows. Section 2 presents the description of the datasets that are used in this paper. Section 3 presents the description and implementation of the proposed methodology. Section 4 presents the comparative results and discussion for the single-user perspective and the multi-user perspective program recommendations. Finally, Section 5 summarizes the outcomes of the paper. It also mentions the limitation of the proposed work and its potential future directions.

2. Description of Datasets

To implement the proposed approach, the three datasets i.e., CelebFaces Attribute Dataset (CelebA), Labeled Faces in the Wild (LFW), and the synthetic dataset is described in Sections 2.1, 2.2, and 2.3, respectively.

2.1. CelebA Dataset

The CelebA dataset [46] consists of huge volumes of face attribute data (i.e., more than 200 K celebrity images). The characteristics such as huge pose variations and background clutter in these images led this dataset to be considered for the proposed approach. Further, the details of quantities, diversities, and annotations in this dataset are given as follows. There are 10,177 identities, 202,599 face images, 5 locations of landmarks, and 40 binary attribute annotations per image.

2.2. LFW-People Dataset

The LFW-People dataset [47] comprises photographs of faces intended for learning the problem of unrestrained face recognition. In this dataset, there are 13,000 face images gathered from the internet; further, the face in each image is labeled with the respective person's name. It also contains two or more distinct images of 1680 people. The summary of the datasets CelebA and LFW-People are given in Table 1.

Table 1. Details of CelebA and LFW-People datasets.

Dataset	Dataset Size
CelebA	202,599 images
LFW-People	13,000 images
Total no. of images	215,599 images

2.3. Synthetic Dataset

To implement the proposed approach, a synthetic dataset that contains the user data, program data, and image data is used. The columns related to user data are 'User ID' and 'User priority' which represent the identity and priority of the users, respectively. The columns related to program data are 'Channels list', 'Programs list', and 'Duration on each channel'. The columns 'Channels list' and 'Programs list' represents the list of real-time channels and programs, respectively that are interesting for a user. The time spent on a particular channel by a user is presented in the column 'Duration on each channel'. The column related to the image data is 'Image' which holds the facial image of the user. The details of the synthetic dataset are given in Table 2, and the respective channels and program details are given in Table 3.

Table 2. Synthetic dataset.

User ID	Channels List	Programs List	Duration on Each Channel	User Priority	Image
1001	▪ Sonypix ▪ Star Movies ▪ Star Sports ▪ India Today	▪ Cricket ▪ Football ▪ News ▪ Action Movies ▪ Reality Show	▪ Sonypix (70 min) ▪ Star Movies (120 min) ▪ Star Sports (132 min)	1	User1.png
1002	▪ Colors ▪ Star Plus ▪ Star World ▪ HBO ▪ Zee TV	▪ Action Movies ▪ Dance Programs ▪ Cookery Program ▪ Reality Show ▪ Cartoon	▪ Colors (62 min) ▪ HBO (98 min) ▪ Star Plus (122 min) ▪ Zee TV (121 min)	2	User2.png

Table 2. *Cont.*

User ID	Channels List	Programs List	Duration on Each Channel	User Priority	Image
1003	▪ Star Gold ▪ Sony Max ▪ Utv Movies ▪ Sonypix	▪ Fantasy Movies ▪ Drama, ▪ Cookery Program ▪ Chat Show ▪ Game Show	▪ Star Gold (98 min) ▪ Sony Max (101 min) ▪ UTV Movies (72 min) ▪ Sonypix (126 min)	3	User3.png
1004	▪ Star Sports2 ▪ Star Utsav ▪ Colors Infinity ▪ Big Magic ▪ Zee Bollywood	▪ Cricket ▪ Football ▪ Cartoon ▪ Documentary	▪ Star Sports2 (102 min) ▪ Star Utsav (68 min) ▪ Colors Infinity (72 min) ▪ Big Magic (58 min) ▪ Zee Bollywood (105 min)	4	User4.png
1005	▪ &Pictures ▪ Bflix Movies ▪ B4u Movies ▪ Colors ▪ Star Sports 2	▪ Action Movies ▪ Horror Movies, ▪ Reality Show ▪ Dance Show ▪ Cricket ▪ Golf	▪ &pictures (98 min) ▪ Bflix Movies (121 min) ▪ B4U Movies (112 min) ▪ Colors (73 min) ▪ Star Sports2 (109 min)	5	User5.png

Table 3. Identities for channels and programs.

Channel ID	Channel Name	Program ID	Program Name
5001	Sonypix	6001	Cricket
5002	Star Movies	6002	Football
5003	Star Sports	6003	News
5004	India Today	6004	Action Movies
5005	Colors	6005	Reality Show
5006	Star Plus	6006	Action Movies
5007	Star World	6007	Dance Programs
5008	HBO	6008	Cookery Program
5009	Zee TV	6009	Reality Show
5010	Star Gold	6010	Cartoon
5011	Sony Max	6011	Fantasy Movies
5012	UTV Movies	6012	Drama
5013	Star Sports2	6013	Chat Show
5014	Star Utsav	6014	Game Show
5015	Colors Infinity	6015	Football
5016	Zee Bollywood	6016	Cartoon

3. Description and Implementation of the Proposed Methodology

The conceptual model of the proposed personalized program recommendation system is shown in Figure 1.

As a preliminary step, the real-world datasets that contain face images are considered as input. The image data in these datasets are trained using the CNN algorithm for feature extraction. Once the image data are trained, the process proceeds to the capturing of user images by using the smart TV camera module. The captured user image is verified with the existing user image in the synthetic dataset. If the captured image is matched, then it proceeds to feature extraction of the captured image. Further, hybrid filtering is applied based on the user perspective, i.e., either single-user or multi-user. The list of personalized programs is recommended as output after the filtering process.

The implementation flow of the proposed personalized program recommendation system is shown in Figure 2. The process starts by reading the face image datasets i.e., CelebA and LFW-People discussed in Section 2. The preprocessing of these datasets is done to remove the noisy data in the face images and make the datasets ready to implement the proposed approach. The preprocessing of image data includes resizing images to the same size, morphological transformation, normalization, and augmentation. The morphological transformation represents modification related to the image shape and size. In this transformation, the steps i.e., erosion, dilation, opening, and closing operations will be performed. Normalization is an important preprocessing step where the pixel values are scaled to fit a specific range. Data augmentation solves the insufficiency of the data to adequately accomplish the classification task.

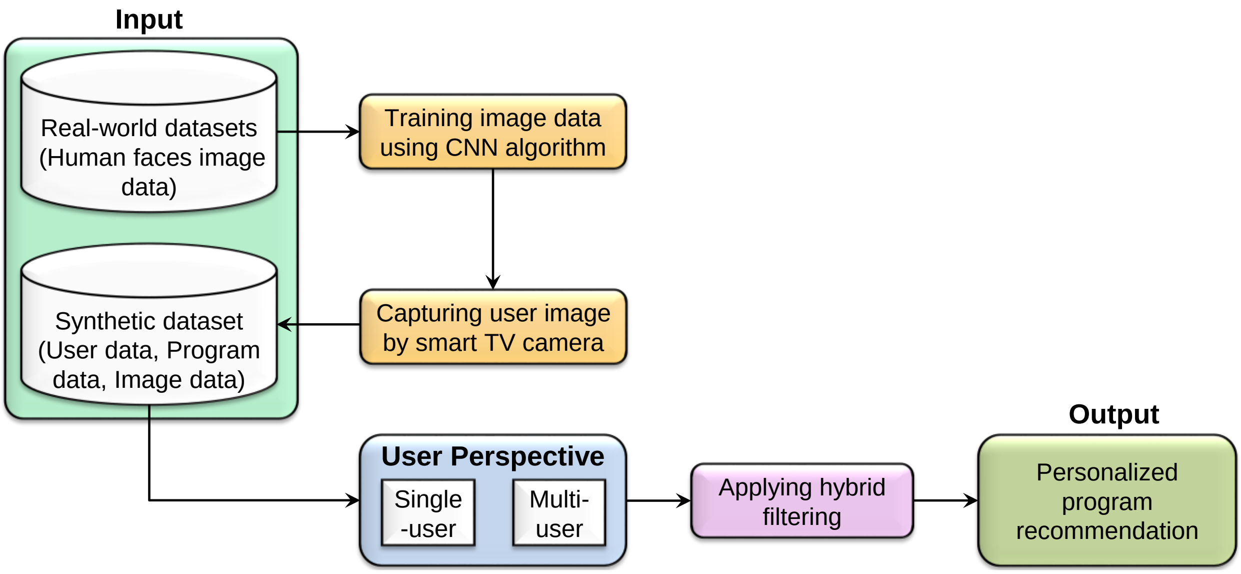

Figure 1. Conceptual model for the personalized program recommendation system.

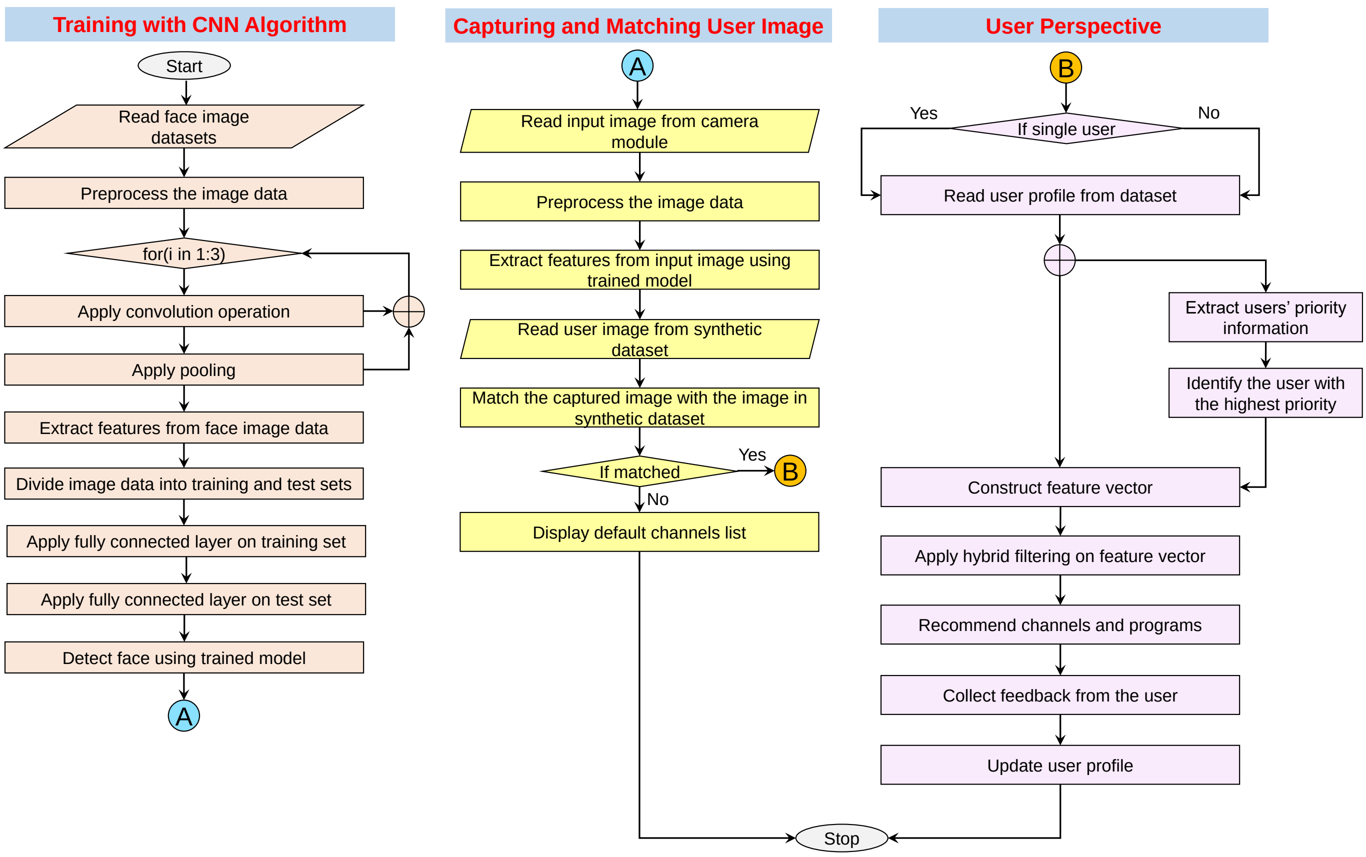

Figure 2. Implementation flow of the proposed personalized program recommendation system.

Once the datasets are preprocessed, then convolution operation and pooling operations are applied iteratively to the face image data for feature extraction. Here, during the convolution operation, the important features of the dataset will be learned. Further, in the next step, i.e., during the pooling operation, the dimensionality of those learned features is reduced, consequently, it reduces the number of parameters to be learned in the network.

The classes in the considered datasets are 'clear face' and 'unclear face'. It is noticed that there is a class imbalance in the datasets. To handle this class imbalance in the datasets, a method named synthetic minority oversampling technique (SMOTE) is applied. After this, the classes in both datasets CelebA and LFW-People are balanced. In the CelebA dataset, it is observed that the images related to the class 'clear face' are 112,343 and the class 'unclear face' are 90256. Similarly, in the LFW-People dataset, it is observed that the images related to the class 'clear face' are 6766 and the class 'unclear face' are 6234. Further, the datasets are divided into a training set (70% of image data), a validation test (10% of data from the training set), and a test set (30% of image data). Additionally, individually consider the total images in both training sets and the total images in both testing sets for the implementation. Apply the fully connected layer on the training set and train the model on the image data. Similarly, apply the fully connected layer on the test set to detect the face using the trained model. Besides, this trained model will be used for detecting the features in the captured user image.

The user image is captured by the camera module of the smart TV. Further, the captured image is preprocessed to remove noisy data in the image. This preprocessed image is given to the trained model for feature extraction. Based on the features, the face in the captured image is detected and then matches this image with the user image in the synthetic dataset. If the image is not matched, then display the default list of the programs and then stop the process. If the image is matched, check the type of the user, i.e., whether single-user or multi-user. If the user is a single user, read the user profile from the dataset and construct the feature vector.

Apply hybrid filtering on the feature vector and recommend the channels and programs. If the user is a multi-user, then extract the user's priority. Based on the priority level, the user is identified. Construct the feature vector and apply hybrid filtering on the feature vector. However, before constructing the feature vector from the user profile, the preprocessing of the user profiles is to be performed. This includes tokenization, stemming, and stop word removal. Recommend the channels and programs. Additionally, the feedback from the user is collected and the user profile is updated accordingly. Section 3.1 describes the CNN algorithm and Section 3.2 describes various filtering approaches corresponding to the recommendation systems.

3.1. CNN Algorithm

The task of the CNN algorithm is to reduce the size of the images into a simpler format to analyze and make a reliable forecast while retaining information. The CNN architecture for the proposed approach is shown in Figure 3. From this, it is inferred that the three layers, namely, the convolution layer, pooling layer, and fully connected layer, act as the building blocks to make up the desired convolutional neural network. The description of these three layers and the CNN's layer configuration is given as follows.

The convolution layer is CNN's backbone and is in charge of carrying out convolution operations. The element in this layer that does the convolution process is the kernel/filter (matrix). The kernel makes horizontal and vertical changes based on the stride rate until the entire image is scanned. Even though the kernel is smaller than a picture, it is deeper. The kernel height and width will be spatially small if the image comprises three (i.e., Red-Green-Blue (RGB)) channels, but its depth will cover all three.

The basic functionality of the pooling layer is to decrease dimensionality. It helps to lessen the computing power that is needed to process the data. Two categories of pooling, such as maximum pooling and average pooling are considered, which can be classified. Maximum (Max) pooling returns the highest value from the kernel-covered region of the image. Average pooling gives the average of all the values in the area of the picture covered by the kernel.

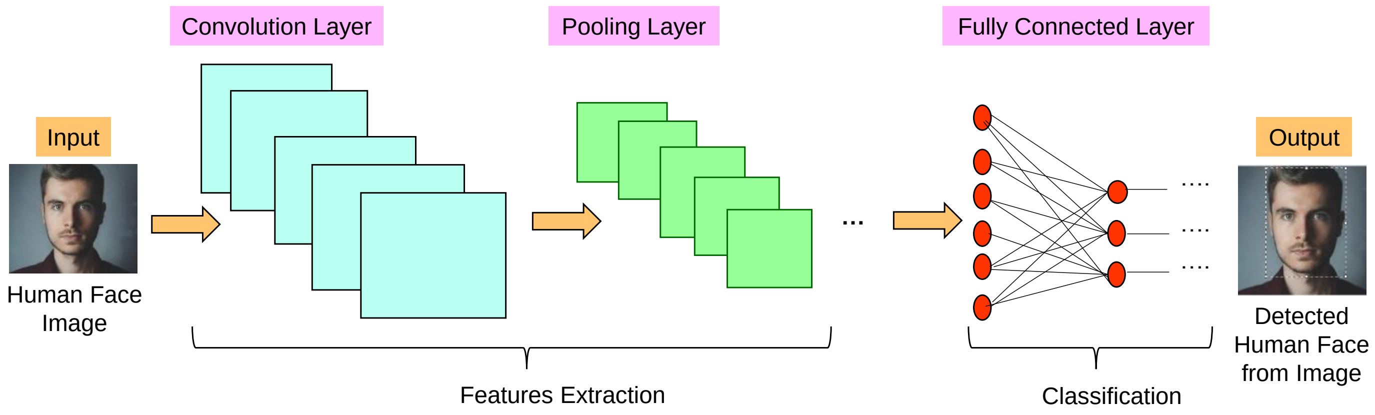

Figure 3. CNN Architecture.

The flattened input used by the fully connected layer means that each input is tied to each neuron. The flattened vector is then transmitted via a few more fully connected layers, where the usual mathematical functional operations are carried out. At this moment, the classification process begins. If fully connected layers are present, they are often located near the end of CNN architectures.

The configuration of layers of the proposed CNN is described in Table 4. In the proposed method, three phases of convolution and pooling layer are used. In the first phase, 64 different kernels of each size (4, 4) matrix are utilized for convolution operation. In the second and third phases of the convolution operation, the same 64 kernels of each size (3, 3) are utilized. The average pooling with size (3, 3) is utilized with every convolution operation for reducing the size of the learned features.

Table 4. Layer configuration of CNN.

Layer Type	Filter/Size	Activation
Conv2D	64 (4, 4)	ReLu
Maxpool	(4, 4)	Nil
Conv2D	64 (3, 3)	ReLu
Conv2D	64 (3, 3)	ReLu
Avgpool2D	(3, 3)	Nil
Conv2D	128 (3, 3)	ReLu
Conv2D	128 (3, 3)	ReLu
Avgpool2D	(3, 3)	Nil
Dense	1024	ReLu
Dropout	Nil	Rate = 0.28
Dense	1024	ReLu
Dropout	Nil	Rate = 0.28
Dense	2	Softmax

The rectified linear unit (ReLu) activation function is used to maintain non-linearity in the output of each convolution operation. The features derived from the given image through CNN are given to the dense neural network. A fully connected neural network is used for detecting the faces from the learning. This neural network is set with two hidden layers each containing 1024 neurons and one output layer with 2 neurons. The activation functions ReLu and Softmax are used in the hidden and output layers, respectively to detect the user faces on the captured image. Here, the learning rate is set as 0.28 to optimize the process during backpropagation. The convolution operation and ReLu activation function equations are given in Equations (1) to (2).

- Convolution Operation:

$$S_{ij} = (I * K)_{i,j} = \sum_{a=0}^{m-1} \sum_{b=0}^{n-1} I_{i+a,j+b} K_{a,b} \tag{1}$$

where, $S_{i,j}$ = Output of convolution on the i^{th} filter and j^{th} input

I = Input image matrix
K = Kernel matrix
i = Filter number
j = Input grid number concerning neuron
a = 0 to m−1, and
b = 0 to n−1

- ReLu Activation:

$$y = max(0, x) \tag{2}$$

where, x = Pixel value of convolution matrix, y = Normalized output.

3.2. Concepts of Filtering Techniques

The basic operation of filtering is to suggest the relevant items. The types of filtering content-based filtering, collaborative filtering, and hybrid filtering are discussed as follows. In content-based filtering, the channel vector is matched with the user profile vector. The matching score is calculated by the dot product of the channel vector and user profile vector as given in Equation (3).

$$SIM(ch, u) = \sum_{i=1}^{d} ch_i u_i \tag{3}$$

where, ch = Chanel vector with programs list, u = User profile vector, i = 1 to d.

In collaborative filtering, the matching profiles of users are considered for the program recommendation. To match the profiles of users, the cosine similarity score between the user profile vectors is calculated as given in Equation (4).

$$SIM(X, Y) = \frac{\sum_{i=1}^{n} X_i * Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2} * \sqrt{\sum_{i=1}^{n} Y_i^2}} \tag{4}$$

where, X and Y = Profile vectors of two users, namely X and Y, i = 1 to n.

The combination of content-based and collaborative filtering techniques is referred to as hybrid filtering as shown in Figure 4. Once the captured image of the user is matched with the existing image in the synthetic dataset, the profile details are retrieved and converted as feature vectors. Collaborative filtering is applied to these feature vectors of currently watching users and the available users to recommend the programs. During the collaborative filtering process, the sparsity issue was faced in the feature vector. This issue was addressed by using the matrix factorization technique across the user profiles. The content-based filtering is applied between user profile vectors and channel vectors of programs to recommend the programs. Hybrid filtering aggregates the results of both collaborative filtering and content-based filtering to recommend the programs.

The algorithmic steps for the hybrid filtering process are given in Algorithm 1. The input for this hybrid filtering is Cont_set (Content-based set) and Coll_set (Collaborative set). The expected output is that the top k items set are denoted by R_k. Initially, arrange the items of Cont_set and Coll_set in descending order based on the similarity score. The similarity scores of items from both Content_set and Coll_set are to be compared and the top k items are added to the set R_k.

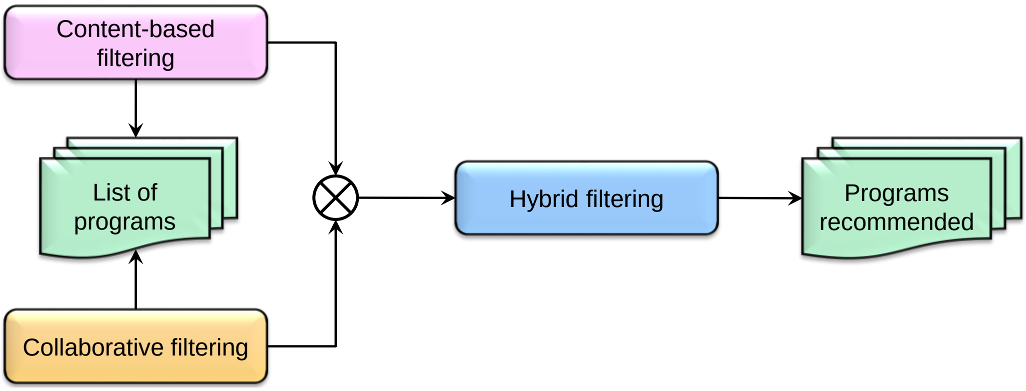

Figure 4. Hybrid filtering.

Algorithm 1. Hybrid filtering process.

Input: Cont_set, Coll_set
Output: top k items set, R_k
Begin
arrange items of Cont_set and Coll_set in descending order based on the similarity score
for each x ∈ Cont_set
    for each y ∈ Coll_set
        if(score(x) > score (y))
$R_k = R_k\ U_x$
else
$R_k = R_k\ U_y$
if size(R_k = = k)
End

However, if there is a situation that the facial information of the smart TV user in the captured image is not properly matched with the existing facial information of the image in the dataset, then the CNN model considers that user as a new user and the system proceeds to the default program playlist without any recommendation.

4. Results and Discussion

The hyperparameters used for implementing the proposed approach are given in Table 5. From this table, it is noticed that the total number of epochs used for the training is 60 with each batch size of 60. Besides, the binary cross entropy is used as a loss function to find the error rate, and the adaptive moment estimation is used as the optimizer to tune the weights during the backpropagation phase.

Table 5. Hyperparameters.

Hyperparameter	Value/Name
Epoch	60
Batch size	60
Loss function	Binary cross entropy
Optimizer	Adaptive moment estimation

The results of the proposed implementation are presented in two sections. Section 4.1. presents the results corresponding to the training CNN algorithm and Section 4.2. presents the results corresponding to the recommendation system.

4.1. Results Corresponding to Training CNN Algorithm

This section presents the training performance of the CNN algorithm on CelebA and LFW-People datasets for the detection of faces. The performance metrics such as precision, recall, and F-measure are computed and compared as shown in Figure 5. From this figure, it is observed that the overall training performance is approximately 95% on both datasets. This highest performance represents that the trained model is outperforming and fit the image data for detecting a human face. A summary of these metrics is given in Table 6. In this table, the Cohen Kappa score is added in addition to the existing measures. This score is used to observe the agreement score between the actual and predicted classes. Usually, this score ranges from −1 to 1. If this score is nearer to 1, then it represents that the trained model is performing well. The Cohen Kappa scores calculated for CelebA and LFW-People datasets are 0.856 and 0.889, respectively. As these scores are nearer to 1, the performance of the trained model is good.

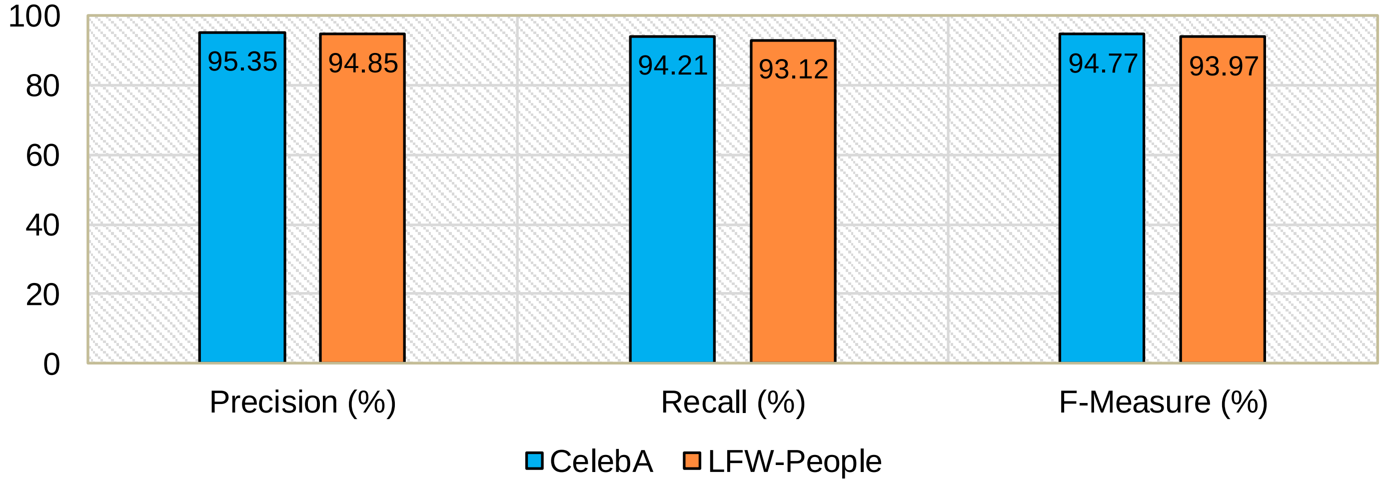

Figure 5. Training performance of CNN algorithm.

Table 6. Performance metrics for training with CNN.

Dataset	Precision (%)	Recall (%)	F-Measure (%)	Cohen Kappa Score (−1 to 1)
CelebA	95.35	94.21	94.77	0.856
LFW-People	94.85	93.12	93.97	0.889

The training accuracy, validation accuracy, training loss, and validation loss of the CNN model on the datasets CelebA and LFW-People are given in Figures 6–9. In these figures, the blue-colored line represents the CelebA dataset and the orange-colored line represents the LFW-People dataset. Further, the number of epochs (0 to 100) is considered on the x-axis and the loss score (0 to 1) is considered on the y-axis. From Figures 6 and 7, it is observed that the accuracy scores of training and validation are gradually increased when the number of epochs is increased. Further, it is observed that these accuracy scores on both datasets are settled and no variation is seen after 60 epochs. From Figures 8 and 9, it is observed that the loss scores of training and validation are gradually decreased when the number of epochs is increased. Further, it is observed that these loss scores on both datasets are settled and no variation is seen after 60 epochs. Hence, it is evident that the performance of the trained CNN model is better after 60 epochs.

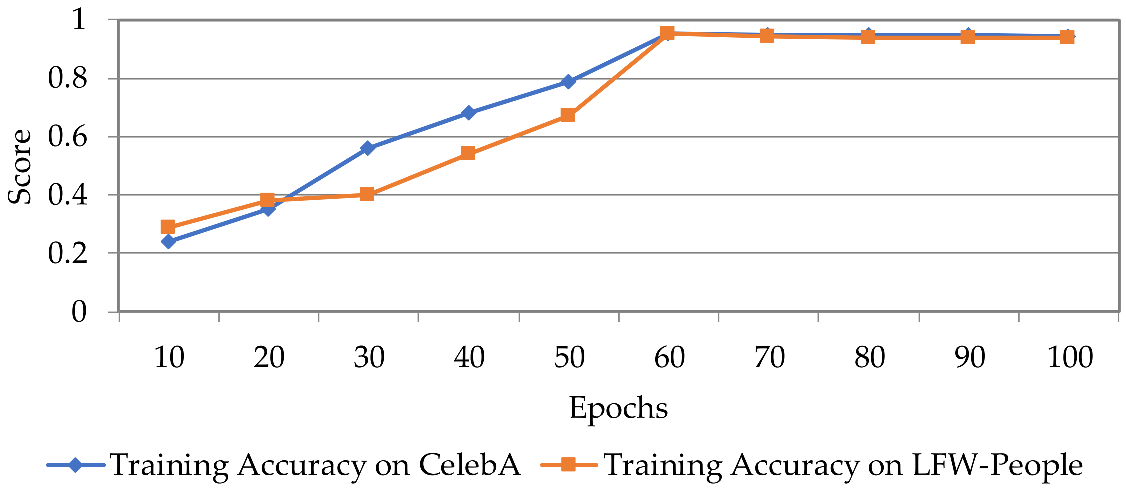

Figure 6. Training accuracy of CNN model on CelebA and LFW-People datasets.

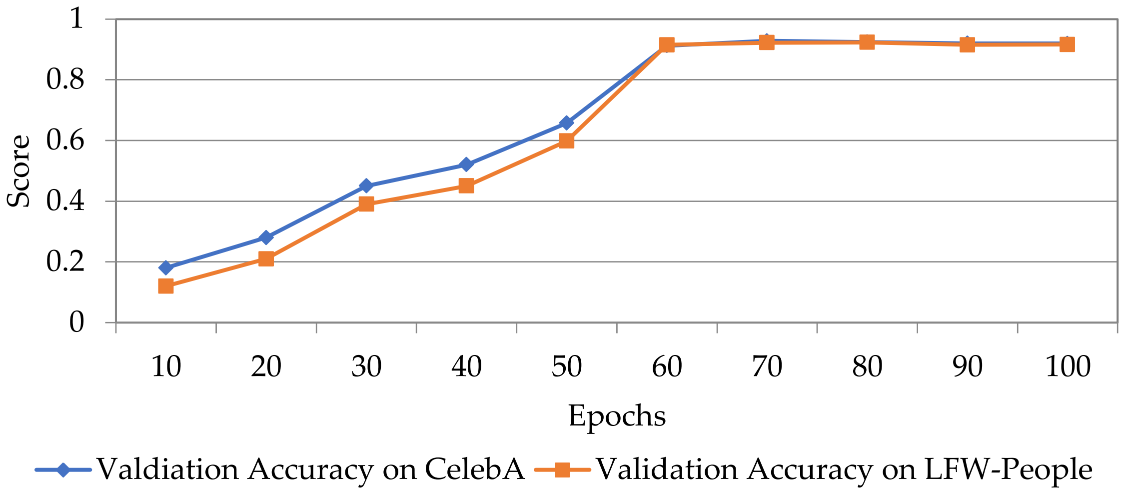

Figure 7. Validation accuracy of CNN model on CelebA and LFW-People datasets.

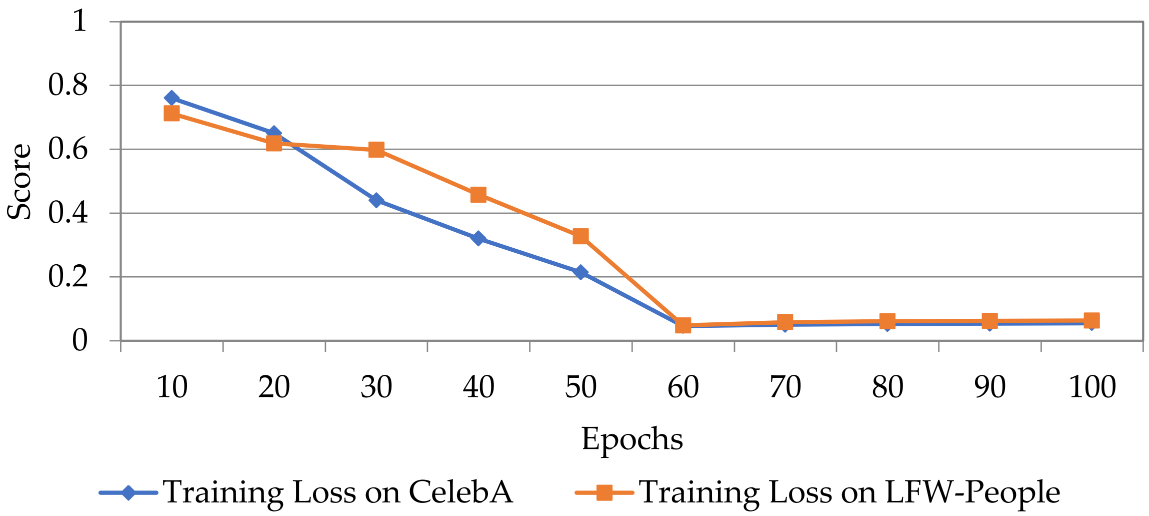

Figure 8. Training loss of CNN model on CelebA and LFW-People datasets.

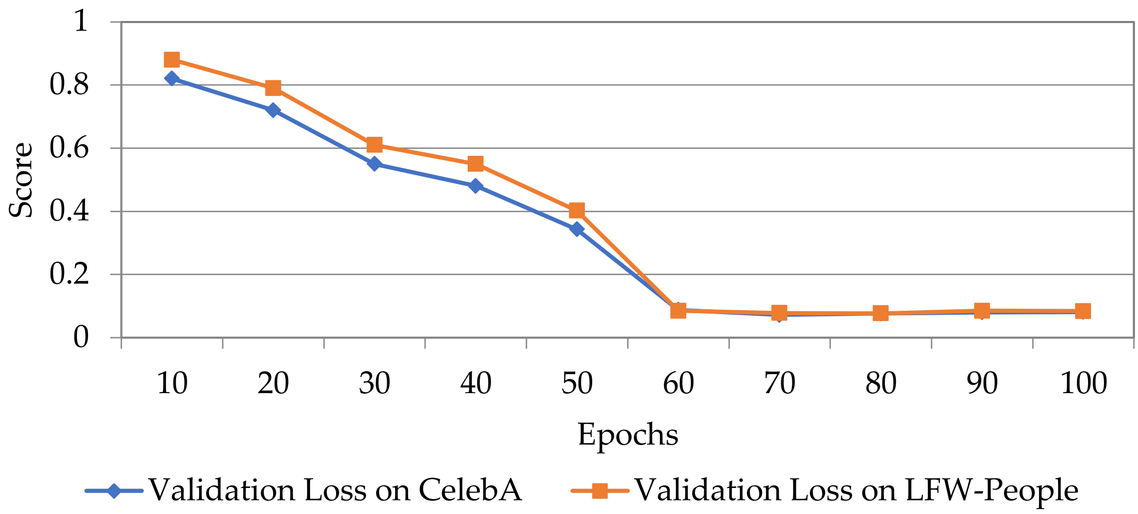

Figure 9. Validation loss of CNN model on CelebA and LFW-People datasets.

4.2. Results Corresponding to Recommendation System

This section discusses the performance of the proposed personalized program recommendation system. Various performance metrics, namely, the area under curve (AUC), precision, and recall are used for the analysis. These metrics are computed from the equations given in Equations (5)–(7).

$$AUC(X) = \frac{1}{|X|(m - |X|)} \sum_{x \in X} \sum_{x' \in (\{1,\ldots,m\})/X} \delta(x < x') \tag{5}$$

$$Prec(X) = \frac{|x \in X : x \leq p|}{p} \tag{6}$$

$$Recall(X) = \frac{|x \in X : x \leq p|}{|X|} \tag{7}$$

where,

X = relevant set
m = size of recommended set
p = top p recommended items
x = relevant channels from the recommended set
x' = irrelevant channels from the recommended set.
if $x > x'$
then $\delta(x > x') = 1$
else $\delta(x > x') = 0$

Further, the program recommendation and performance comparison of filtering techniques in a recommendation system for a single-user perspective and multi-user perspective are discussed in Sections 4.2.1 and 4.2.2, respectively.

4.2.1. Personalized Program Recommendation and Performance Comparison of Filtering Techniques: (Single-User Perspective)

The details of the program recommendation for single users are given in Table 7. From this table, it is evident that the program preferences for user 1001 are 4, and the recommended programs by the proposed recommendation system are 12. From this recommendation, it is observed that 10 programs are matched with the programs of user interest. The majority of users' interests are correctly matching with the recommended programs.

Table 7. Details of the personalized program recommendation for a single user.

User ID	Program Preferences	Recommended Programs	Matching Programs
1001	4	12	10
1002	5	10	8
1003	5	11	7
1004	4	8	7
1005	6	13	11

Further, based on the programs recommended by the system from a single-user perspective, the performance comparison of various filtering techniques (namely content-based, collaborative, and hybrid) is given. From the comparison shown in Figure 10; it is observed that the hybrid filtering technique outperformed the other two filtering techniques (content-based and collaborative). The performance metrics computed with this hybrid filtering are 83.78% of AUC, 86.78% of precision, and 85.32% of recall, which are superior to other techniques. Similarly, the performance metrics computed with collaborative filtering are 79.56% of AUC, 80.56% of precision, and 79.54% of recall. The performance metrics computed with the content-based filtering are 78.32% of AUC, 79.32% of precision, and 77.43% of recall. A summary and comparison of these metrics are given in Table 8.

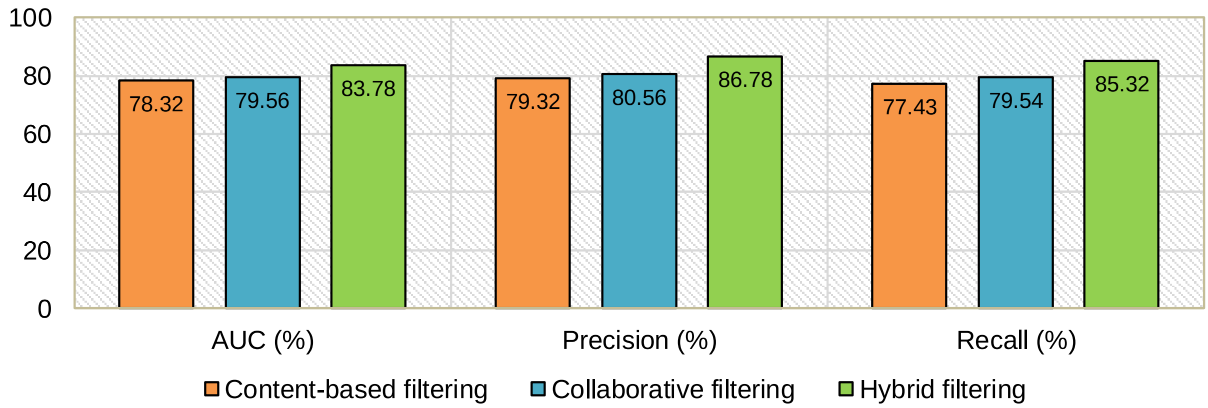

Figure 10. Performance comparison of filtering techniques in single-user program recommendation.

Table 8. Performance comparison of filtering techniques in single-user program recommendation.

Method	AUC (%)	Precision (%)	Recall (%)
Content-based filtering	78.32	79.32	77.43
Collaborative filtering	79.56	80.56	79.54
Hybrid filtering	83.78	86.78	85.32

4.2.2. Personalized Program Recommendation and Performance Comparison of Filtering Techniques: (Multi-User Perspective)

The details of the program recommendation for multiple users are given in Table 9.

Table 9. Details of personalized program recommendations for multiple users.

Group ID	Group Size	Common Program Preferences	Recommended Programs	Matching Programs
2001	5	6	15	10
2002	4	8	13	9
2003	6	12	19	16
2004	3	10	11	10
2005	7	8	21	18

In this Table 9, it is observed that the users are assigned a group identity of different sizes. Each group has some common program preferences. For example, the users with 'Group ID' 2001 and 'Group size' of five have six programs as a common preference. The proposed recommendation system recommended 15 programs to this group and out of them, 10 match with the users' interests. Similarly, the programs are recommended to other groups of users.

Further, based on the programs recommended by the system from a multi-user perspective, the performance comparison of various filtering techniques (namely content-based, collaborative, and hybrid) is given. From the comparison shown in Figure 11, it is observed that the hybrid filtering technique outperformed the other two filtering techniques (content-based and collaborative). The performance metrics computed with this hybrid filtering are 80.17% of AUC, 82.68% of precision, and 81.57% of recall, which are superior to other techniques. Similarly, the performance metrics computed with collaborative filtering are 75.36% of AUC, 76.21% of precision, and 74.91% of recall. The performance metrics computed with the content-based filtering are 74.32% of AUC, 73.32% of precision, and 72.34% of recall. A summary and comparison of these metrics are given in Table 10.

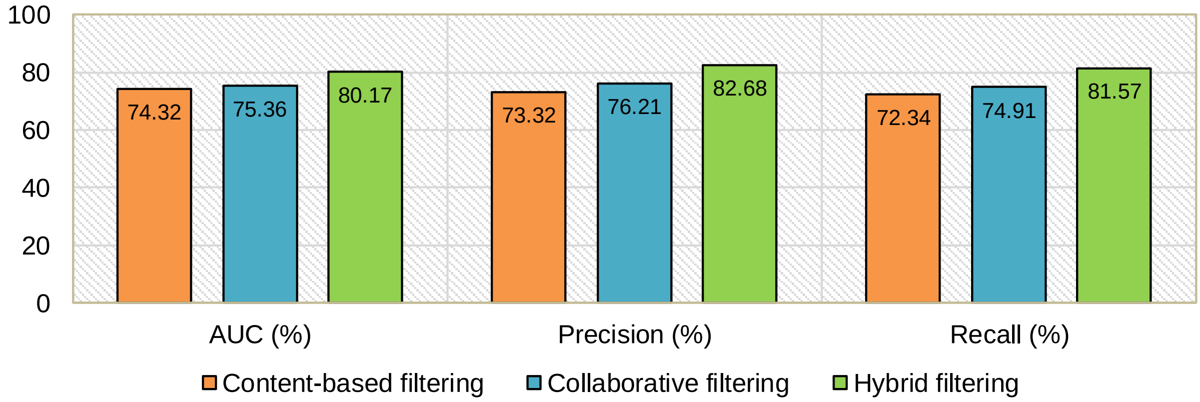

Figure 11. Performance comparison of filtering techniques in multi-user program recommendation.

Table 10. Performance comparison of filtering techniques in multi-user program recommendation.

Method	AUC (%)	Precision (%)	Recall (%)
Content-based filtering	74.32	73.32	72.34
Collaborative filtering	75.36	76.21	74.91
Hybrid filtering	80.17	82.68	81.57

5. Conclusions

This paper proposes a CNN-based personalized program recommendation system for smart television users based on the type of user (single-user and multi-user) and their priorities and interests. The overall training performance of the proposed CNN algorithm on both the CelebA and LFW-people datasets is approximately 95%. This performance showcases that the trained model effectively fits the image data in detecting the human face. The significant observations made from this proposed approach are given as follows.

From a single-user perspective, the performance of the proposed hybrid filtering technique outperformed the content-based and collaborative filtering techniques. The performance of the proposed approach in terms of the metrics AUC (83.78%), precision (86.78%), and recall (85.32%) are obtained as higher than the conventional content-based and collaborative filtering techniques.

From a multi-user perspective, the performance of the proposed hybrid filtering technique outperformed the content-based and collaborative filtering techniques. The performance of the proposed approach in terms of the metrics AUC (80.17%), precision (82.68%), and recall (81.57%) are obtained as higher than the conventional content-based and collaborative filtering techniques.

Hence, the proposed CNN-based personalized program recommendation system has successfully helped in recommending the programs to smart TV users considering both single-user and multi-user perspectives effectively.

5.1. Limitation

The limitation of the present work is the image processing of twins and the recommendation of the programs to their interests. When there are twins before the camera module with different program interests, it is difficult to recommend personalized programs.

5.2. Future Work

The future work of the recommendation systems will be a voice-based interactive smart TV, which can interact with the user to run personalized programs based on the user's interests.

Author Contributions: Conceptualization, K.V.D.; Data curation, K.V.D., Y.V.P.K., S.I.S., M.I.M.B., H.S., P.P.K., A.F. and A.T.A.; Formal analysis, K.V.D., Y.V.P.K., S.I.S., M.I.M.B., H.S., P.P.K., A.F. and A.T.A.; Funding acquisition, H.S.; Investigation, K.V.D., H.S., P.P.K. and M.I.M.B.; Methodology, K.V.D., Y.V.P.K., S.I.S., M.I.M.B., H.S., P.P.K., A.F. and A.T.A.; Project administration, A.F. and A.T.A.; Software, K.V.D.; Supervision, A.T.A.; Validation, S.I.S., A.F., Y.V.P.K., P.P.K., S.I.S., H.S. and Y.V.P.K.; Visualization, A.F., Y.V.P.K., S.I.S., M.I.M.B. and A.T.A.; Writing—original draft, M.I.M.B. and P.P.K.; Writing—review and editing, K.V.D., Y.V.P.K., S.I.S., M.I.M.B., H.S., P.P.K., A.F. and A.T.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding. The APC was funded by VIT-AP University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank VIT-AP University, Amaravati, Andhra Pradesh, India for funding the open-access publication fee for this research work. Further, the authors would like to thank Prince Sultan University, Riyadh, Saudi Arabia. Special acknowledgments are given to Automated Systems and Soft Computing Lab (ASSCL), Prince Sultan University, Riyadh, Saudi Arabia for allowing this collaborative research work with VIT-AP University.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

AI	artificial intelligence
AUC	area under curve
CelebA	celebfaces attribute
CNN	convolutional neural network
LFW	labeled faces in the wild-people
Max	maximum
ReLu	rectified linear unit
RGB	red-green-blue
SMOTE	synthetic minority oversampling technique
TV	television

References

1. Kasaraneni, P.P.; Yellapragada, V.P.K. A Systematic Approach for Exploration, Behavior Analysis, and Visualization of Redundant Data Anomalies in Smart Home Energy Consumption Dataset. *Int. J. Renew. Energy Res.* **2022**, *12*, 109–123.
2. Kasaraneni, P.P.; Yellapragada, V.P.K. Analytical Approach to Exploring the Missing Data Behavior in Smart Home Energy Consumption Dataset. *J. Renew. Energy Environ.* **2022**, *9*, 37–48.
3. Kasaraneni, P.P.; Yellapragada, V.P.K. Systematic Statistical Analysis to Ascertain the Missing Data Patterns in Energy Consumption Data of Smart Homes. *Int. J. Renew. Energy Res.* **2022**, *12*, 1560–1573.

4. Kasaraneni, P.P.; Yellapragada, V.P.K. Simple and Effective Descriptive Analysis of Missing Data Anomalies in Smart Home Energy Consumption Readings. *J. Energy Syst.* **2021**, *5*, 199–220.
5. Kasaraneni, P.P.; Yellapragada, V.P.K.; Moganti, G.L.K.; Flah, A. Analytical Enumeration of Redundant Data Anomalies in Energy Consumption Readings of Smart Buildings with a Case Study of Darmstadt Smart City in Germany. *Sustainability* **2022**, *14*, 10842. [CrossRef]
6. Bures, M.; Macik, M.; Ahmed, B.S.; Rechtberger, V.; Slavik, P. Testing the Usability and Accessibility of Smart TV Applications Using an Automated Model-Based Approach. *IEEE Trans. Consum. Electron.* **2020**, *66*, 134–143. [CrossRef]
7. Kim, T.; Choi, S.; Bahn, H. A Personalized Interface for Supporting Multi-Users in Smart TVs. *IEEE Trans. Consum. Electron.* **2016**, *62*, 310–315. [CrossRef]
8. Alanazi, A.; Alseid, M. Artificial Intelligence Based Recommendation System for Analyzing Social Bussiness Reviews. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 122–132. [CrossRef]
9. Zhang, Q.; Lu, J.; Jin, Y. Artificial Intelligence in Recommender Systems. *Complex Intell. Syst.* **2021**, *7*, 439–457. [CrossRef]
10. Li, X. Research on Reform and Breakthrough of News, Film, and Television Media Based on Artificial Intelligence. *J. Intell. Syst.* **2022**, *31*, 992–1001. [CrossRef]
11. Portugal, I.; Alencar, P.; Cowan, D. The Use of Machine Learning Algorithms in Recommender Systems: A Systematic Review. *Expert Syst. Appl.* **2018**, *97*, 205–227. [CrossRef]
12. Kulkarni, S.; Rodd, S.F. Context Aware Recommendation Systems: A Review of the State of the Art Techniques. *Comput. Sci. Rev.* **2020**, *37*, 100255. [CrossRef]
13. Abbas, A.; Zhang, L.; Khan, S.U. A Survey on Context-Aware Recommender Systems Based on Computational Intelligence Techniques. *Computing* **2015**, *97*, 667–690. [CrossRef]
14. Hassanien, A.E.; Tolba, M.; Azar, A.T. Advanced Machine Learning Technologies and Applications. In Proceedings of the Second International Conference, AMLTA 2014, Cairo, Egypt, 28–30 November 2014; Proceedings, Communications in Computer and Information Science. Springer: Berlin/Heidelberg, Germany; Volume 488, ISBN 978-3-319-13460-4.
15. Azar, A.T.; Hassanien, A.E.; Kim, T.H. Expert System Based On Neural-Fuzzy Rules for Thyroid Diseases Diagnosis. In Proceedings of the International Conference on Bio-Science and Bio-Technology (BSBT 2012), Gangneug, Korea, 16–19 December 2012; Springer: Berlin/Heidelberg, Germany, 2012; Volume 353, Communications in Computer and Information Science series. pp. 94–105, ISBN 978-3-642-35520-2. [CrossRef]
16. Aziz, A.S.A.; Hassanien, A.E.; Azar, A.T.; Hanafy, S.E. Genetic Algorithm with Different Feature Selection Techniques for Anomaly Detectors Generation. In Proceedings of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), Kraków, Poland, 8–11 September 2013.
17. Jothi, G.; Inbarani, H.H.; Azar, A.T. Hybrid Tolerance Rough Set: PSO Based Supervised Feature Selection for Digital Mammogram Images. *Int. J. Fuzzy Syst. Appl. (IJFSA)* **2013**, *3*, 15–30. [CrossRef]
18. Jothi, G.; Inbarani, H.H.; Azar, A.T.; Devi, K.R. Rough set theory with Jaya optimization for acute lymphoblastic leukemia classification. *Neural Comput. Appl.* **2019**, *31*, 5175–5194. [CrossRef]
19. Anter, A.M.; Azar, A.T.; El-Bendary, N.; Hassanien, A.E.; Abu ElSoud, M. Automatic Computer Aided Segmentation for Liver and Hepatic Lesions Using Hybrid Segmentations Techniques. In Proceedings of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), Kraków, Poland, 8–11 September 2013.
20. Emary, E.; Zawbaa, H.; Hassanien, A.E.; Schaefer, G.; Azar, A.T. Retinal Vessel Segmentation based on Possibilistic Fuzzy c-means Clustering Optimised with Cuckoo Search. In Proceedings of the IEEE 2014 International Joint Conference on Neural Networks (IJCNN 2014), Beijing, China, 6–11 July 2014.
21. Saad, A.; Fouad, H.; Mohamed, A.A. Situation-Aware Recommendation System for Personalized Healthcare Applications. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 1–15. [CrossRef]
22. Dhelim, S.; Chen, L.; Aung, N.; Zhang, W.; Ning, H. A Hybrid Personality-Aware Recommendation System Based on Personality Traits and Types Models. *J. Ambient. Intell. Humaniz. Comput.* **2022**, *13*, 1–14. [CrossRef]
23. Lin, C.Y.; Chen, H.S. Personalized Channel Recommendation on Live Streaming Platforms. *Multimed. Tools Appl.* **2019**, *78*, 1999–2015. [CrossRef]
24. Li, M.; Wang, L. A Survey on Personalized News Recommendation Technology. *IEEE Access* **2019**, *7*, 145861–145879. [CrossRef]
25. Wu, F.; Lyu, C.; Liu, Y. A Personalized Recommendation System for Multi-Modal Transportation Systems. *Multimodal Transp.* **2022**, *1*, 100016. [CrossRef]
26. Pantano, E.; Priporas, C.V.; Stylos, N.; Dennis, C. Facilitating Tourists' Decision Making through Open Data Analyses: A Novel Recommender System. *Tour. Manag. Perspect.* **2019**, *31*, 323–331. [CrossRef]
27. Laaroussi, S.; Baataoui, A.; Halli, A.; Satori, K. Item-to-item Recommender System with Simultaneous Use of Multiple Images for Image Mosaicking Creation in Dynamic Scenes. *IET Image Process* **2020**, *14*, 4726–4735. [CrossRef]
28. Ahmed, U.; Lin, J.C.W.; Srivastava, G.; Djenouri, Y. A Nutrient Recommendation System for Soil Fertilization Based on Evolutionary Computation. *Comput. Electron. Agric.* **2021**, *189*, 106407. [CrossRef]
29. Islek, I.; Oguducu, S.G. A Hierarchical Recommendation System for E-Commerce Using Online User Reviews. *Electron. Commer. Res. Appl.* **2022**, *52*, 101131. [CrossRef]
30. Shuvo, S.S.; Yilmaz, Y. Home Energy Recommendation System (HERS): A Deep Reinforcement Learning Method Based on Residents' Feedback and Activity. *IEEE Trans. Smart Grid* **2022**, *13*, 2812–2821. [CrossRef]

31. Luo, F.; Ranzi, G.; Kong, W.; Dong, Z.Y.; Wang, S.; Zhao, J. Non-intrusive Energy Saving Appliance Recommender System for Smart Grid Residential Users. *IET Gener. Transm. Distrib.* **2017**, *11*, 1786–1793. [CrossRef]
32. Mekouar, L.; Iraqi, Y.; Damaj, I.; Naous, T. A Survey on Blockchain-Based Recommender Systems: Integration Architecture and Taxonomy. *Comput. Commun.* **2022**, *187*, 1–19. [CrossRef]
33. Gu, K.; Fan, Y.; Di, Z. How to Predict Recommendation Lists That Users Do Not Like. *Phys. A Stat. Mech. Its Appl.* **2020**, *537*, 122684. [CrossRef]
34. Silveira, T.; Zhang, M.; Lin, X.; Liu, Y.; Ma, S. How Good Your Recommender System Is? A Survey on Evaluations in Recommendation. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 813–831. [CrossRef]
35. Alam, I.; Khusro, S.; Khan, M. Personalized Content Recommendations on Smart TV: Challenges, Opportunities, and Future Research Directions. *Entertain. Comput.* **2021**, *38*, 100418. [CrossRef]
36. Alam, I.; Khusro, S. Tailoring Recommendations to Groups of Viewers on Smart TV: A Real-Time Profile Generation Approach. *IEEE Access* **2020**, *8*, 50814–50827. [CrossRef]
37. Zhu, X.; Guo, J.; Li, S.; Hao, T. Facing Cold-Start: A Live TV Recommender System Based on Neural Networks. *IEEE Access* **2020**, *8*, 131286–131298. [CrossRef]
38. Topalli, İ.; Kilinç, S. User Profiling for TV Program Recommendation Based on Hybrid Television Standards Using Controlled Clustering with Genetic Algorithms and Artificial Neural Networks. *Turk. J. Electr. Eng. Comput. Sci.* **2020**, *28*, 1567–1583. [CrossRef]
39. Jang, J.; Yi, M.Y. Determining and Validating Smart TV UX Factors: A Multiple-Study Approach. *Int. J. Hum. -Comput. Stud.* **2019**, *130*, 58–72. [CrossRef]
40. Alam, I.; Khusro, S.; Khan, M. Factors Affecting the Performance of Recommender Systems in a Smart TV Environment. *Technologies* **2019**, *7*, 41. [CrossRef]
41. Kim, J. Recommendation Method of Smart TV Programs Reflecting Content Consumption Concentration Calculation. *J. Eng.* **2020**, *2020*, 444–447. [CrossRef]
42. Krstic, M.; Bjelica, M. Personalized Program Guide Based on One-Class Classifier. *IEEE Trans. Consum. Electron.* **2016**, *62*, 175–181. [CrossRef]
43. Pyo, S.; Kim, E.; Kim, M. LDA-Based Unified Topic Modeling for Similar TV User Grouping and TV Program Recommendation. *IEEE Trans. Cybern.* **2015**, *45*, 1476–1490.
44. Malakar, S.; Ghosh, M.; Bhowmik, S.; Sarkar, R.; Nasipuri, M. A GA Based Hierarchical Feature Selection Approach for Handwritten Word Recognition. *Neural Comput. Appl.* **2020**, *32*, 2533–2552. [CrossRef]
45. Milosevic, S.; Bezdan, T.; Zivkovic, M.; Bacanin, N.; Strumberger, I.; Tuba, M. Feed-Forward Neural Network Training by Hybrid Bat Algorithm. In *Modelling and Development of Intelligent Systems*; Simian, D., Stoica, L.F., Eds.; Springer International Publishing: New York, NY, USA, 2021.
46. CelebA Dataset. Available online: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html (accessed on 10 November 2022).
47. LFW-People Dataset. Available online: https://www.kaggle.com/datasets/atulanandjha/lfwpeople?resource=download (accessed on 10 November 2022).