Gender Classification using Machine Learning

C.Nagalakshmi[1], Dr.Naga Deepthi Ponnaganti[2]
[1]PG Scholar, Department of CSE, Sir C.R.Reddy College of Engineering, Eluru.
[2]Associate Professor, PG Scholar, Department of CSE, Sir C.R.Reddy College of Engineering, Eluru

Abstract: Gender classification is a crucial task in various applications such as security, marketing, and healthcare. In this project, we propose a Convolutional Neural Network (CNN) approach for automated gender classification from facial images. We preprocess the dataset to extract facial features and normalize pixel values. The CNN architecture consists of convolutional layers for feature extraction and fully connected layers for classification. We augment the dataset to increase robustness and reduce overfitting. We train the model using backpropagation and optimize it using gradient descent. We evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score. Experimental results demonstrate the effectiveness of our approach in accurately classifying gender from facial images. Our model achieves competitive performance compared to existing methods, indicating its potential for real-world applications. The proposed CNN-based gender classification system provides a reliable and efficient solution for automated gender recognition tasks.

Keywords:- CNN, Gameclassification, AI, ML

I. INTRODUCTION

Overview of the Project

Gender classification, the process of determining the gender of an individual based on various visual cues, holds significant importance in a wide array of fields including security, marketing, healthcare, and human-computer interaction. The ability to automatically identify gender from visual data such as facial images is not only essential for practical applications like surveillance and access control but also facilitates personalized user experiences and targeted advertising. With the advancements in computer vision and machine learning, particularly the emergence of Convolutional Neural Networks (CNNs), the accuracy and efficiency of gender classification systems have witnessed notable improvements.

This introduction serves to provide an overview of the importance of gender classification, the challenges associated with this task, and the motivation behind employing CNNs for automated gender recognition. Additionally, it outlines the objectives and structure of this project aimed at developing a robust CNN-based gender classification model.

The significance of gender classification in various domains cannot be overstated. In security and surveillance systems, accurate gender classification is crucial for identifying potential threats or monitoring demographic patterns in public spaces. Similarly, in marketing and advertising, understanding the gender distribution of target audiences enables businesses to tailor their campaigns more effectively and enhance customer engagement. In healthcare, gender classification can aid in personalized treatment plans and medical diagnostics, where certain conditions may exhibit gender- specific symptoms or prevalence rates. Moreover, gender recognition plays a vital role in human-computer interaction applications, enabling devices and interfaces to adapt to users' gender preferences and behaviors.

Despite its significance, gender classification from visual data presents several challenges. Facial images are highly variable due to factors such as pose variations, lighting conditions, facial expressions, and occlusions. Additionally, cultural and ethnic diversity further complicates the task, as certain facial features may exhibit different characteristics across populations. Moreover, the presence of implicit biases in datasets and models can lead to inaccuracies and disparities

in gender classification results, highlighting the importance of fairness and equity in algorithmic decision-making systems.

The motivation behind utilizing CNNs for automated gender classification stems from their remarkable success in various computer vision tasks, particularly image classification and object recognition. CNNs are well-suited for learning hierarchical representations of visual features, enabling them to capture complex patterns and relationships within the data. By leveraging convolutional layers, max-pooling operations, and non-linear activation functions, CNNs can effectively extract discriminative features from facial images, making them ideal candidates for gender recognition tasks.

Deep learning techniques are now widely used for various tasks such as classification, automatic feature extraction, object recognition etc., due to their high classification accuracy. Motivated from other fields, researchers have also exploited deep learning methods for gender prediction and classification from facial images. The following paragraphs present a summary of some studies.

Janahiraman and Subramaniam (Janahiraman and Subramaniam, 2019) aimed to make gender classification using different models of CNN architecture. A dataset was created from facial images from Malaysians and some Caucasians people. Their results achieved accuracy of 88% with the VGG-16 model, 85% with the ResNet-50 model and 49% with the Mobile Net model. Akbulut et al (Akbulut et al., 2014) performed gender recognition from facial images using Local Recipient Areas-Excessive Learning Machine (LRA- ELM) and CNN architecture. The experiments were carried out using approximately 11 thousand images from the Adience dataset for age and gender recognition (Eidinger et al., 2014). The proposed method resulted in 80% and 87.13% accuracy with LRA-ELM and CNN respectively.

In (Gündüz and Cedimoğlu,2019), a comparative analysis was performed among propose method, Alex Net and VGG-16 models for gender classification including women, men, old, young, children, and babies. Experimental results show accuracy of 72.20% with the proposed CNN model, 99.41% with the VGG-16 model, and 65.63% with the AlexNet model.

Arora and Bhatia (Arora and Bhatia, 2018) proposed a CNN model for gender classification with facial images. In experimental studies, 1500 images were used for training and 1000 images were selected from the CASIA database and verified. As a result of the experiments, 98.5% accuracy was achieved.

II. LITERATURE SURVEY

Deep Learning Approach to Predict the Gender of pedestrians:

Raza et al. (2018) introduced a deep learning approach to predict the gender of pedestrians. Their method involved a preprocessing step to segment pedestrians from images, followed by the utilization of stacked autoencoders with softmax classifiers for classification. Notably, they achieved accuracy rates of 82.9%, 81.8%, and 82.4% for anterior, posterior, and mixed views in the MIT dataset, respectively. Furthermore, they attained approximately 91.5% accuracy in the PETA dataset. This study showcases the effectiveness of deep learning techniques in gender classification tasks, particularly in pedestrian images, demonstrating promising results across different viewpoints and datasets.

An experimental comparison of gender classification methods.

To date, limited surveys have been conducted on human gender recognition and classification. Mäkinen et al. [9] conducted a comparative study focusing on gender classification methods integrated with automatic real-time face detection. Their research aimed to evaluate the performance and efficacy of various gender classification techniques in conjunction with face detection systems. By analyzing different methodologies, Mäkinen et al. provided insights into the strengths and limitations of existing approaches for gender recognition tasks. However, despite this effort, comprehensive surveys specifically dedicated to the field of human gender recognition remain scarce. The study by Mäkinen et al. contributes valuable insights into the state- of-the-art techniques for gender classification, particularly in real-time scenarios where accurate and efficient face detection is essential. Nonetheless, there remains a need for further research and surveys to comprehensively explore advancements and challenges in human gender recognition.

SEXNET: A Neural Network Identifies Sex From Human Faces

Research on gender classification using facial images began in the early 1990s. Golomb et al. [24] introduced a multi-layer neural network approach for gender classification from human faces in 1990. Employing a Cottrell-style back-propagation image compression network, they achieved a gender classification error rate of 8.1%. This pioneering study laid the groundwork for subsequent research in the field, demonstrating the potential of neural networks for automated gender recognition from facial features. Golomb et al.'s work marked a significant milestone in the development of gender classification methodologies, setting the stage for further advancements in computer vision and machine learning techniques for gender analysis.

Golomb BA, Lawrence DT, Sejnowski TJ. Sexnet: A neural network identifies sex from human faces. Advance in Neural Information Processing Systems

Since the 1990s, Gender Recognition and Classification have been pivotal in research and development. Golomb et al. [1] pioneered using multi-layer neural networks for gender classification. They manually aligned facial images and compressed around 900 unit images into 40 for experimentation. Despite these challenges, their approach reported an 8.1% error rate, showcasing the early promise of neural networks in gender classification. Golomb et al.'s innovative work laid the groundwork for subsequent advancements, emphasizing the significance of gender classification in various domains. This pioneering study continues to influence and inspire further research in the field of computer vision and machine learning.

Identifying gender from unaligned facial images by set classification

Chu. et al. [2] conducted a study tackling gender classification with unaligned face images, focusing on single faces from which various poses were cropped and combined into sets. They converted these image sets into subspaces and utilized correlation coefficients to measure similarity between subspaces. Discriminant analysis of Canonical Correlation (DCC) was then employed to determine gender accurately. Their research utilized the FERET and MORPH face databases for experimentation. This innovative approach demonstrated the potential of utilizing correlation coefficients and discriminant analysis techniques for gender classification tasks, contributing to the ongoing evolution of methodologies in the field of computer vision and pattern recognition.

Ameneh Shobeirinejad, Yongsheng Gao. Gender classification using interlaced derivative patterns.

Shobeirinejad and Gao [3] introduced the Interlaced Derivative Pattern (IDP) method for facial feature extraction, generating a feature vector by capturing distinctive facial features. IDP constructs a four-channel derivative image representing directional information at angles of 0°, 45°, 90°, and 135°. This approach enhances gender face recognition by incorporating crucial details from different orientations, thus providing a comprehensive representation of facial characteristics. The utilization of IDP enriches the feature extraction process, offering a more robust and informative approach for gender classification tasks in computer vision and pattern recognition research.

Dataset:

This project utilizes an image dataset specifically designed for gender classification tasks. The dataset consists of numerous individual images, each containing a human face. These faces represent a diverse range of ages, ethnicities, and appearances.

Data Labeling

Each image within the dataset is accompanied by a corresponding label. These labels typically fall into two categories: "male" and "female." The labels are crucial for training a machine learning model to identify the gender depicted in a given image.

Data Splitting

The dataset is often divided into distinct subsets for training, validation, and testing purposes. The training set constitutes the bulk of the data and is used to train the model to recognize gender patterns. The validation set helps fine-tune the model's hyperparameters, while the testing set evaluates the model's performance on unseen data.

Data Preprocessing

Before feeding the images into a machine learning model, some preprocessing steps might be necessary. These steps could involve resizing images to a uniform size, converting them to grayscale for consistency, or applying normalization techniques to ensure data values fall within a specific range.

Dataset Considerations

It's important to acknowledge potential biases within the dataset. The source and composition of the images can influence the model's performance on specific demographics. Additionally, ethical considerations regarding privacy and fairness should be addressed when utilizing these datasets for real-world applications.

SYSTEM DESIGN

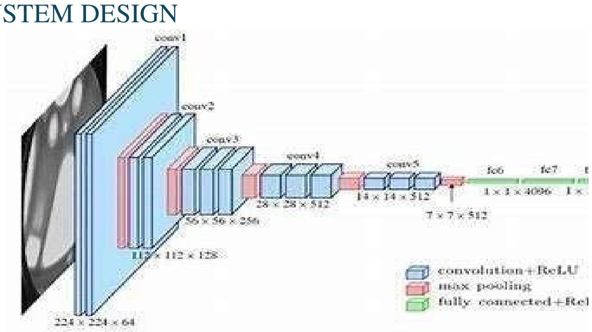


Figure 1: CNN Architecture:

Convolutional Neural Networks (CNNs) are a specialized class of deep neural networks, particularly effective in image processing tasks. Their architecture is inspired by the organization of the animal visual cortex, designed to automatically and adaptively learn spatial hierarchies of features from input images.

Dataset Overview

Understanding the dataset is paramount. The gender_dataset_face dataset likely comprises images of faces, each annotated with gender labels. Prior to model construction, preprocessing steps such as normalization, resizing, and possibly data augmentation might be required to standardize image dimensions and improve model robustness.

Input Layer:

The input layer represents the raw pixel values of the input images. Each pixel serves as an input neuron to the network. The size of this layer is determined by the dimensions of the input images in the gender_dataset_face dataset.

Convolutional Layers

Convolutional layers are fundamental to CNNs. They consist of multiple learnable filters that convolve across input images, extracting features such as edges, textures, and patterns. Each filter produces a feature map, and stacking multiple convolutional layers enables the network to learn hierarchical representations of visual features.

Activation Functions

Activation functions, like ReLU (Rectified Linear Unit), introduce non-linearity into the network, enabling it to model complex relationships within the data. ReLU is commonly used due to its simplicity and effectiveness in mitigating the vanishing gradient problem.

Pooling Layers

Pooling layers reduce the spatial dimensions of feature maps generated by convolutional layers. Common pooling operations include max pooling and average pooling, which down sample feature maps, reducing computational complexity and spatial information while retaining essential features.

Batch Normalization

Batch normalization normalizes the activations of each layer within a mini-batch during training. It improves network stability and accelerates convergence by reducing internal covariate shift, which helps prevent gradients from vanishing or exploding during training.

Dropout Regularization

Dropout regularization is a technique to prevent overfitting by randomly deactivating a fraction of neurons during training. This encourages the network to learn more robust features and prevents reliance on specific neurons, thus improving generalization performance.

Fully Connected Layers

Fully connected layers, or dense layers, process the flattened output from preceding layers. They learn complex relationships between features and are typically employed in the latter part of the network architecture, leading to the final classification decision.

Output Layer

The output layer represents the predicted gender labels. For binary classification tasks, a single neuron with a sigmoid activation function is often used. For multi-class classification, the number of neurons equals the number of classes, with softmax activation providing probability distributions over classes.

Loss Function

The loss function quantifies the difference between predicted and actual labels. For binary classification tasks, binary cross-entropy is commonly used, while categorical cross-entropy is used for multi-class classification. The goal during training is to minimize this loss.

Optimizer

The optimizer adjusts the model parameters during training to minimize the loss function. Adam, RMSprop, and SGD with momentum are popular optimizers. Learning rate, a crucial hyperparameter, governs the step size during optimization.

Training Process

During training, the CNN is fed batches of images from the dataset. Backpropagation computes gradients of the loss function with respect to network parameters, and the optimizer updates parameters to minimize loss. This iterative process continues until convergence or a predefined stopping criterion is met.

Evaluation Metrics

Evaluation metrics such as accuracy, precision, recall, and F1-score assess model performance. Accuracy measures the proportion of correctly classified instances, while precision and recall focus on the correctness and completeness of predictions, respectively. F1-score balances precision and recall.

Hyperparameter Tuning

Hyperparameters like learning rate, batch size, dropout rate, and network architecture significantly impact model performance. Grid search, random search, or more advanced techniques like Bayesian optimization are employed to find optimal hyperparameter configurations.

Methodology

Data Preprocessing

Data preprocessing is an essential step in any machine learning project, including gender classification. It involves preparing and cleaning the dataset to ensure it is suitable for training. Common preprocessing techniques include resizing images to a consistent size, normalizing pixel values to a standardized scale, and removing any irrelevant or noisy data points. Additionally, data augmentation methods such as rotation, flipping, and cropping may be applied to increase the diversity of the dataset. By carefully preprocessing the data, we can improve the performance and generalization ability of the gender classification model.

Image preprocessing

Data preprocessing is crucial for a gender classification project to ensure the quality and consistency of input data. This typically involves several steps, including resizing images to a uniform size, normalizing pixel values to a standardized scale, and possibly applying techniques like data augmentation to enhance model generalization. Additionally, the dataset may require cleaning to remove noisy or irrelevant images, and labels should be encoded properly for model compatibility. By preprocessing the data meticulously, potential biases and inconsistencies can be mitigated, resulting in improved model performance and robustness in gender classification tasks.

Image resizing and cropping:

Image resizing involves adjusting the dimensions of images to a uniform size, ensuring consistency in the dataset. This process is crucial for CNNs as they require fixed-size inputs. Cropping, on the other hand, involves selecting a specific region of interest within an image, discarding unnecessary background. It helps focus the model on relevant features while reducing computational overhead. Both resizing and cropping are vital preprocessing steps to enhance the effectiveness of image-based machine learning tasks like gender classification.

CNN Components

CNN model consists of different types of components:

Kernel / Filter / Feature extractor

Stride

Padding

Pooling

Flattening

Kernel

Kernel is also known as a filter or feature detector because it will detect the features from the input image. Kernel is represented in a matrix form; it will move over the input image by using stride value given and gives the output as a dot product by considering sub-region of input data.

Formula to calculate the output matrix size after doing convolution

$$O = \left\lfloor \frac{i-k}{s} \right\rfloor + 1$$

i = input matrixsize k = kernel matrix sizes = stride value

O = output matrix size

Example

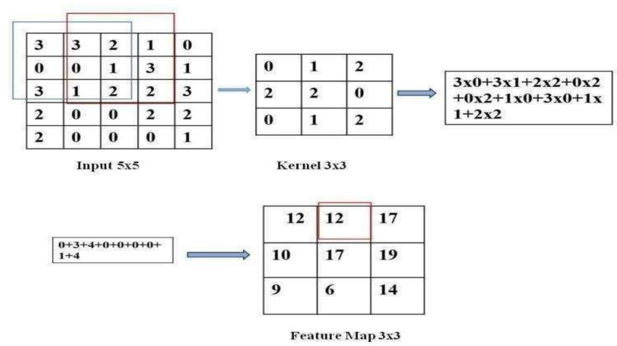
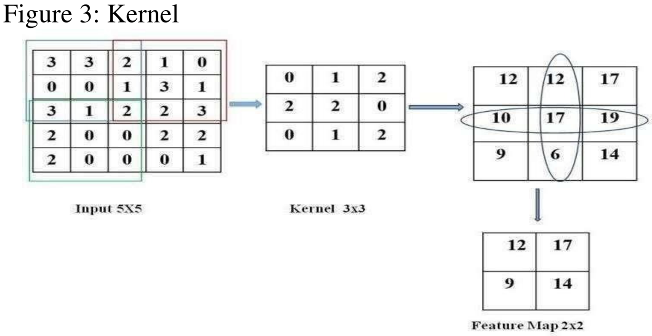


Figure 3: Kernel



Stride

The amount of filter movement is called stride. It moved across the input image from

left to right, top to bottom, with one-pixel value change on the horizontal position and one-pixel changeinvertical position. The default value of stride in 1-D is 1 and in 2-D is (1,1) for height and width movement.

Example:

Fig 6. stride

Figure 2: Padding

Padding is mainly used for solving the Border Problem. It is used when the edge pixels play’s an important role in classification task. If an input image is padded and then it passed into the CNN model to give the more accurate analysis of images.

Formula

$$O = \left\lfloor \frac{i-k+2p}{s} \right\rfloor + 1$$

O = output matrix sizei = input matrix size k = kernel matrix sizes = stride value

Example

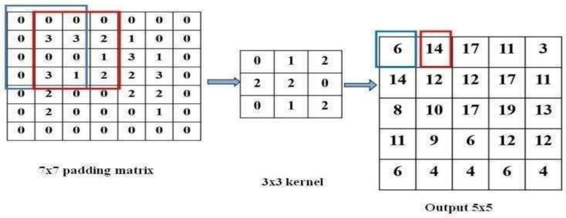


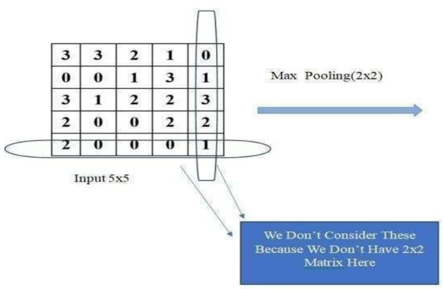
Figure 4: padding

Pooling

Pooling is one of the important components that makes CNN very effective. Pooling is used to reduce the size of input image size in order to reduce the computation power. It has two types of operations.

Max pooling

In max pooling it takes the max value from the sub matrix selected and places that max valuein theoutput matrix.

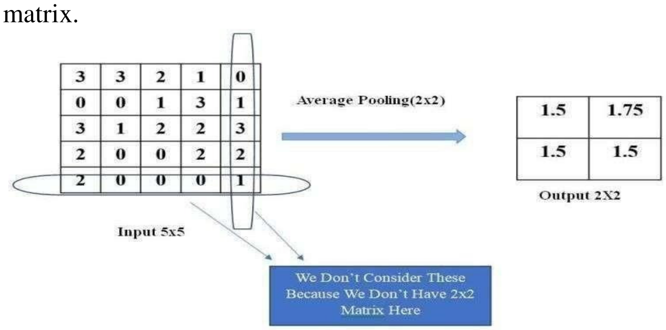


Example

Figure 5: Max Pooling

Average pooling

In average pooling it calculates the average value of the sub matrix selected from the inputmatrixand places in the output matrix.



Example

Figure 6: Average pooling

Flattening

A pooled feature map obtained before the flattening layer is passed into the flattening layer in orderto make it into a single column matrix which is then given as an input to the neural network for processing.

Dropout

Dropout is the method of randomly ignoring the neurons. This method is mainly used to prevent the overfitting problem. Overfitting means that the neurons the co-dependent of each other. For exampledropout (0.2) means it randomly ignores the 20 percent of neurons from the fully connected .

Activation Function

Activation functions play a crucial role in neural networks by deciding whether a neuron should be activated or not. This decision is based on whether the neuron's input information is relevant for the given prediction task. An activation function performs this role by transforming the input signal into an output signal, making it ready for the next layer in the network. The choice of activation function affects the network's ability to converge and the speed of convergence; it can also influence the ability of the network to model complex relationships in the data by introducing non-linearity. Without activation functions, a neural network would simply perform a linear transformation of its input data, limiting its capacity to learn complex patterns [4].

We use activation functions because they help neural networks learn and represent complex data patterns. By introducing non-linear properties to the network, activation functions enable the network to perform tasks beyond those that can be achieved with linear operations, such as classifying non-linearly separable data. This non-linearity is crucial for deep learning models, which stack multiple layers of neurons to understand high-level features in data. Activation functions also help in controlling the flow of gradients during the training process, which is essential for updating the weights of the neurons in a manner that minimizes the loss function.

Types of Activation Function

Sigmoid
Tanh
Relu
Softmax

Sigmoid:

It is a function which is plotted as 'S' shaped graph.
Equation : A = 1/(1 + e-x)
Nature : Non-linear.
Value Range : 0 to 1
Uses : Usually used in output layer of a binary classification, where result is either 0 or 1, as value for sigmoid function lies between 0 and 1 only so, result can be predicted easily to be 1 if value is greater than 0.5 and 0 otherwise.

Tanh activation function:

The activation that works almost always better than sigmoid function is Tanh function also known as Tangent Hyperbolic function. It's actually mathematically shifted version of the sigmoid function. Both are similar and can be derived from each other Equation :

$$f(x) = tanh(x) = \frac{2}{1+e^{-2x}} - 1$$

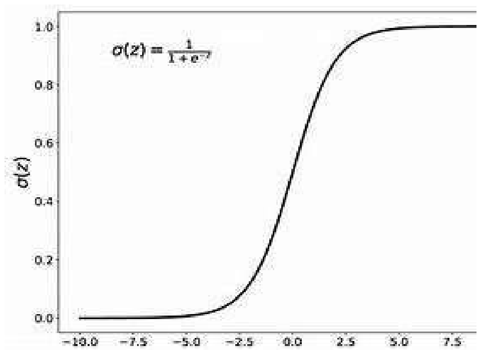


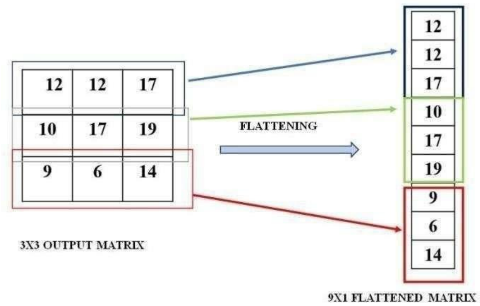
Figure 7: Flatened

Nature : Non-linear.

Figure 8: sigmoid activation function

Value Range :- -1 to +1

Uses :- Usually used in hidden layers of a neural network as it's values lies between -1 to 1 hence the mean for the hidden layer comes out be 0 or very close to it, hence helps in centering the data by bringing mean close to 0.



Example

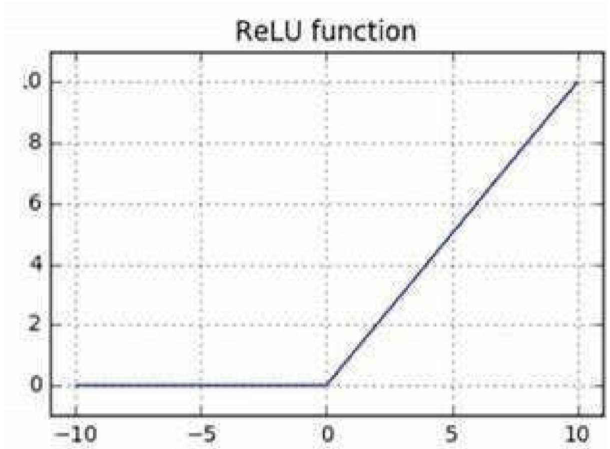


Figure 9: Relu activation function:

It Stands for Rectified linear unit. It is the most widely used activation function. Chiefly implemented in hidden layers of Neural network.

Equation :- A(x) = max(0,x). It gives an output x if x is positive and 0 otherwise.

Value Range :- [0, inf)

Nature :- non-linear, which means we can easily backpropagate the errors and have multiple layers of neurons being activated by the ReLU function.

Uses :- ReLu is less computationally expensive than tanh and sigmoid because it involves simpler mathematical operations. At a time only a few neurons are activated making the network sparse making it efficient and easy for computation.

In simple words, RELU learns much faster than sigmoid and Tanh function.

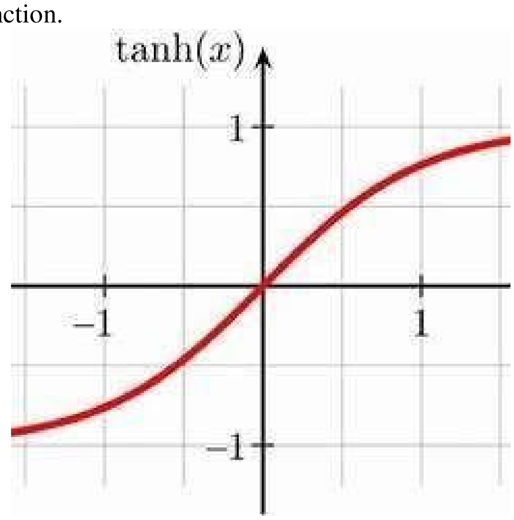


Figure 10: Relu activation

Softmax activation function:

It Stands for Rectified linear unit. It is the most widely used activation function. Chiefly implemented in hidden layers of Neural network.

Equation :- A(x) = max(0,x). It gives an output x if x is positive and 0 otherwise.

Value Range :- [0, inf)

Nature :- non-linear, which means we can easily backpropagate the errors and have multiple layers of neurons being activated by the ReLU function.

Uses :- ReLu is less computationally expensive than tanh and sigmoid because it involves simpler mathematical operations. At a time only a few neurons are activated making the network sparse making it efficient and easy for computation.

In simple words, RELU learns much faster than sigmoid and Tanh function.

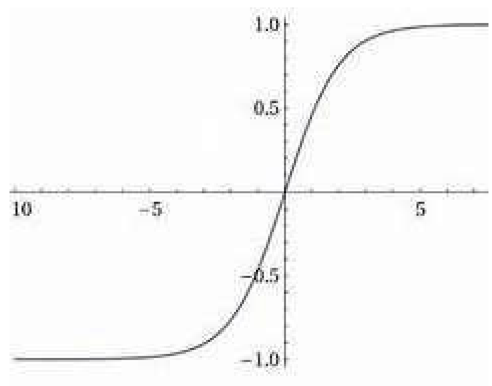


Figure 11: Softmax activation

Flattening

A pooled feature map obtained before the flattening layer is passed into the flattening layer in orderto make it into a single column matrix which is then given as an input to the neural network for processing.

Example

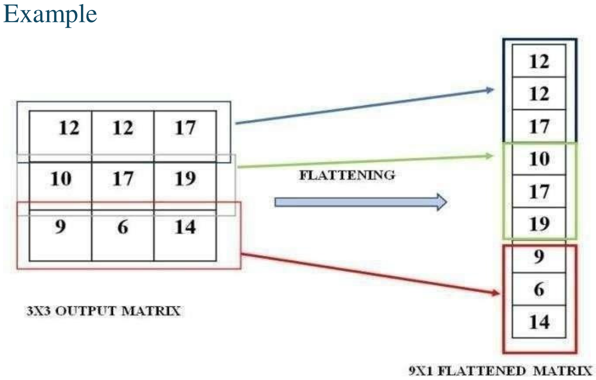


Figure 12: Flattening layer

Dropout

Dropout is the method of randomly ignoring the neurons. This method is mainly used to prevent the overfitting problem. Overfitting means that the neurons the co-dependent of each other. For exampledropout (0.2) means it randomly ignores the 20 percent of neurons from the fully connected network.

CNN Model in System

In the proposed hybrid model for the gender classification project, the CNN plays a pivotal role in discerning gender based on features extracted from preprocessed facial images.

Specifically, the CNN model is activated when facial features indicating gender are discernible. It processes preprocessed facial images to learn distinctive features crucial for gender classification.

Comprising convolutional and pooling layers, the CNN progressively extracts intricate features from input images. These layers analyze facial features, identifying patterns relevant to gender classification. The CNN model's output is a probability distribution across gender classes, aiding subsequent decision-making.

Following CNN processing, a classifier, such as a decision tree, utilizes the probability distribution to assign gender labels, distinguishing between male and female classifications accurately.

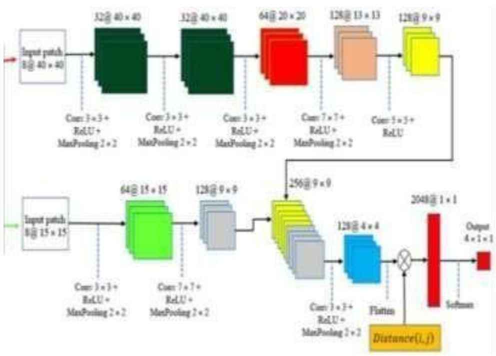


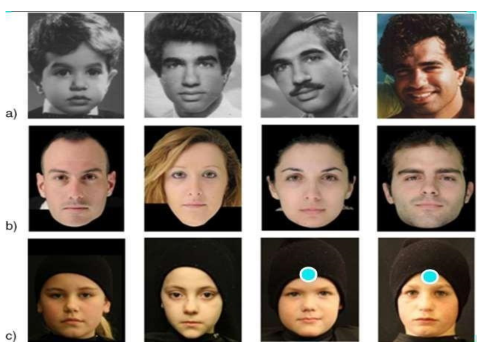
Figure 13: CNN Architecture

International Journal of Computer Science and Mechatronics
A peer reviewed international journal | Article available at http://ijcsm.in | ICI-JIF 8.051
©smsamspublications.com | Vol.10.Issue.3.2024.ISSN: 2455-1910

The CNN model [22] is trained on the preprocessed facial images and corresponding gender labels in the gender_dataset_face dataset to accurately classify gender based on the extracted features. The model undergoes tuning using the validation set to optimize its hyperparameters and ensure robust generalization to unseen data.

Convolutional Neural Networks (CNNs) excel in processing images, comprising multiple layers of neurons that progressively extract intricate features from input images. In the proposed hybrid model for the gender classification project on the gender_dataset_face dataset, the CNN model discerns gender based on features extracted from preprocessed facial images.

The CNN model takes preprocessed facial images as input, extracting meaningful features pivotal for gender classification. Facial images, typically represented in 2D, are processed through layers of convolutional and pooling operations, enabling extraction of increasingly complex features.



The initial layer of the CNN model typically involves convolutional operations applying filters to the input images. Each filter performs convolution by sliding over the input image, computing dot products with pixel values in the receptive field. Resulting feature maps depict filter responses to input images.

Convolutional layer outputs undergo non-linear activation, often utilizing Rectified Linear Unit (ReLU) functions, introducing non-linearity. Subsequently, pooling layers reduce feature map dimensionality by aggregating adjacent pixels. Pooling operations can vary, commonly employing max-pooling or average-pooling techniques.

The output of the pooling layer is then forwarded to subsequent convolutional and pooling layers, extracting increasingly intricate features from the input facial images. This iterative process continues across multiple layers until the model generates a condensed representation of the input, suitable for classification.

The CNN model for gender classification is conventionally trained via backpropagation, involving gradient computation of the loss function with respect to model parameters. Optimization algorithms like stochastic gradient descent (SGD) update parameters to minimize the loss. Cross- entropy loss typically measures the disparity between predicted and true gender labels, optimized during training on preprocessed facial images and corresponding gender labels from the gender_dataset_face dataset. The model undergoes tuning using a validation set to optimize hyperparameters, ensuring robust generalization to unseen data.

The CNN model outputs a probability distribution across gender classes. A classifier, such as a decision tree, utilizes this distribution to classify gender accurately, distinguishing between male and female classifications. Additionally, the decision-making process may incorporate estimated gender size information generated by auxiliary models, enhancing classification accuracy and efficiency.

In summary, the CNN model [23] serves as a cornerstone in the proposed hybrid model for gender classification, facilitating precise gender identification based on features extracted from preprocessed facial images. Integration with auxiliary models and decision-making mechanisms enhances the model's efficacy in gender classification tasks.

The advantages of using CNNs for this project include their ability to learn hierarchical features from large datasets, their ability to generalize to new data, and their ability to handle complex and high-dimensional data such as medical images. With appropriate hyperparameter tuning and architecture design, CNNs can achieve state-of-the-art performance in image segmentation tasks.

CNNs excel in image segmentation tasks due to their ability to automatically learn relevant features for distinguishing different structures in images. This capability makes them suitable for segmenting various anatomical structures and pathologies in medical imaging. By analyzing image patterns, CNNs can accurately delineate regions of interest, aiding in diagnosis and treatment planning. Their adaptability to diverse datasets and ability to capture intricate details make CNNs invaluable tools in medical image analysis, facilitating precise segmentation of organs, lesions, and abnormalities for improved healthcare outcomes.

CNNs offer a significant advantage in end-to-end training, allowing the entire model to be trained directly from raw input

International Journal of Computer Science and Mechatronics

A peer reviewed international journal | Article available at http://ijcsm.in | ICI-JIF 8.051
©smsamspublications.com | Vol.10.Issue.3.2024.ISSN: 2455-1910

data alongside corresponding ground truth segmentation masks. This approach simplifies the segmentation process by removing the requirement for manual feature engineering or extensive preprocessing steps. By learning directly from data, CNNs autonomously discern relevant features and relationships, optimizing segmentation performance. This streamlined training process enhances efficiency and accuracy, enabling CNNs to effectively tackle diverse segmentation tasks across various domains, including medical imaging, computer vision, and more.

Another advantage of CNNs is that they can be easily adapted to handle different types of imaging modalities. This makes them a versatile tool for medical image analysis, as they can be used to segment a wide range of different types of images.

CNNs are also well-suited to handling large amounts of data, which is often necessary for medical image segmentation tasks. Because CNNs can be parallelized across multiple GPUs or CPUs, they can be used to process large datasets in a relatively short amount of time.

Robustness to variations: CNNs are designed to be robust to variations in the input data such as translation, rotation, scaling, and deformation. This makes them effective at handling complex data such as images that may have variations due to differences in imaging protocols.

Hierarchical representations: CNNs can learn hierarchical representations of the input data, with higher layers learning more abstract and complex features. This makes them well-suited to tasks such as image segmentation, where a pixel-level understanding of the image is required.

Transfer learning: Pre-trained CNNs can be used as a starting point for new tasks, with the lower layers of the network serving as a feature extractor for the new data. This can reduce the amount of training data required and improve generalization performance.

Finally, CNNs have been shown to achieve state-of-the-art performance on a wide range of image segmentation tasks, including segmentation of gender, brain tumors, lung cancer, and retinal vasculature, among others. This makes them a promising tool for improving the accuracy and efficiency of classifiying the gender.

RESULTS

Random images for gender classification

Epoch	Train Loss	Training Accuracy	Val Loss	Val Accuracy
1	0.6095	0.6715	0.5179	0.8433
2	0.5150	0.7578	0.4462	0.8200
3	0.4099	0.8256	0.3444	0.8633
4	0.3282	0.8704	0.2295	0.9000
5	0.2383	0.9144	0.1776	0.9400
6	0.1755	0.9422	0.1221	0.9433
7	0.1277	0.9659	0.0631	0.9767
8	0.0838	0.9789	0.0668	0.9767
9	0.0608	0.9848	0.0382	0.9833
10	0.0484	0.9911	0.0505	0.9800

Table 1: Performance of CNN Model for 10 Epochs

CONCLUSION

In conclusion, the implementation of a gender classification project using a webcam presents both opportunities and challenges. Leveraging real-time facial recognition technology offers the potential for seamless integration into various applications, from personalized user experiences to security and surveillance systems. However, ensuring accuracy, fairness, and privacy protection remains paramount. Additionally, addressing challenges such as variability in lighting conditions, facial expressions, and pose variations is crucial for achieving robust performance. Through continued research and development, alongside careful consideration of ethical and technical considerations, gender classification using webcams can contribute to advancements in computer vision and enhance user interactions in diverse domains.

REFERENCES

1. Abdalrady, N. A., and Aly, S., 2020, February, Fusion of Multiple Simple Convolutional Neural Networks for Gender Classification, In 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), IEEE, 251-256.
2. Akbulut, Y., Şengür, A., and Ekici, S., 2017, Gender recognition from face images with deep learning, In 2017 International artificial intelligence and data processing symposium (IDAP), IEEE, 1-4.
3. Arora, S., and Bhatia, M. P. S., 2018, A robust approach for gender recognition using deep learning, In 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, 1-6.

International Journal of Computer Science and Mechatronics

A peer reviewed international journal | Article available at http://ijcsm.in | ICI-JIF 8.051 ©smsamspublications.com | Vol.10.Issue.3.2024.ISSN: 2455-1910

4. Eidinger, E., Enbar, R., and Hassner, T., 2014, Age and gender estimation of unfiltered faces, IEEE Transactions on Information Forensics and Security, 9(12), 2170-2179.
5. Gonzalez, R. C., Woods, R. E., and Eddins, S. L., 2004, Digital Image Processing Using Matlab, 1st Indian Reprint.
6. Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y., 2016, Deep learning Vol. 1, No. 2, Cambridge: MIT press.
7. Gündüz, G., and Cedimoğlu, İ. H., 2019, Derin Öğrenme Algoritmalarını Kullanarak Görüntüden Cinsiyet Tahmini, Sakarya University Journal of Computer and Information Sciences, 2(1), 9-17.
8. He, K., Zhang, X., Ren, S., and Sun, J., 2016, Deep residual learning for image recognition, In Proceedings of the IEEE conference on computer vision and pattern recognition, 770-778.
9. Janahiraman, T. V., and Subramaniam, P., 2019, Gender Classification Based on Asian Faces using Deep Learning, In 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET), 84-89.
10. KAGGLE, 2018, https://www.kaggle.com/ttungl/adience-benchmark-gender-and-age-classification. Sahra Tilki | Gender. Classif. Using DL. Technique VOLUME 02, NO: 01. 2021 131
11. Krizhevsky, A., Sutskever, I., and Hinton, G. E., 2017, ImageNet classification with deep convolutional neural networks, Communications of the ACM, 60(6), 84-90.
12. LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P., 1998, Gradient-based learning applied to document recognition, Proceedings of the IEEE, 86(11), 2278-2324.
13. Levi, G., and Hassner, T., 2015, Age and gender classification using convolutional neural networks, In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 34-42.
14. Ranjan, R., Patel, V. M., and Chellappa, R., 2017, Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition, IEEE transactions on pattern analysis and machine intelligence, 41(1), 121-135.
15. Raza, M., Sharif, M., Yasmin, M., Khan, M. A., Saba, T., and Fernandes, S. L., 2018, Appearance based pedestrians' gender recognition by employing stacked auto encoders in deep learning, Future Generation Computer Systems, 88, 28-39.
16. Şeker, A., Diri, B., and Balık, H. H., 2017, Derin öğrenme yöntemleri ve uygulamaları hakkında bir inceleme, Gazi Mühendislik Bilimleri Dergisi (GMBD), 3(3), 47-64. (1)Akkar, S. and Bommer, J. J., 2006, Influence of long-period filter cut-off on elastic spectral displacements, Earthquake Engineering and Structural Dynamics, 35(9), 1145-1165.
17. Simonyan, K., and Zisserman, A., 2014, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.
18. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R., 2014, Dropout: a simple way to prevent neural networks from overfitting, The journal of machine learning research, 15(1), 1929-1958.
19. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... and Rabinovich, A., 2015, Going deeper with convolutions, In Proceedings of the IEEE conference on computer vision and pattern recognition, 1-9.
20. Yu, Z., Shen, C., and Chen, L., 2017, Gender classification of full body images based on the convolutional neural network, In 2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), IEEE, 707-711.
21. Zeiler, M. D., & Fergus, R., 2014, Visualizing and understanding convolutional networks, In European conference on computer vision, 818-833, Springer.
22. Dedeepya, P., Karishma, D., Manuri, S. G., Raghuvaran, T., Shariff, V., & Sindhura, S. (2023, September 20). Enhancing Cyber Bullying Detection Using Convolutional Neural Network. 2023 4th International Conference on Smart Electronics and Communication (ICOSEC).https://doi.org/10.1109/icosec58147.2023.10276007
23. Arava, Karuna, Chiranjeevi Paritala, Vahiduddin Shariff, S. Phani Praveen, and A Madhuri. 2022. "A Generalized Model for Identifying Fake Digital Images through the Application of Deep Learning." 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), August. https://doi.org/10.1109/icesc54411.2022.9885341.