Real Time Gender and Age Detection

A COURSE PROJECT REPORT

18CSC305J - ARTIFICIAL INTELLIGENCE

Submitted by

Sarath chandra M [RA2011033010160]
Amit kumar [RA2011033010150]
Prince kumar [RA2011033010145]

Under the guidance of

Dr. T. R. Sarvavanan

Associate Professor, Department of Computational Intelligence

in partial fulfillment for the award of the

course of

BACHELOR OF TECHNOLOGY
in

COMPUTER SCIENCE & ENGINEERING
of
FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

S.R.M. Nagar, Kattankulathur, Chengalpattu District
MAY 2023

SRM INSTITUTE OF SCIENCE ANDTECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that Course project report titled **"Real Time Age and Gender Detection"** is the bonafide work of **Prince Kumar (RA2011033010145), Sarath chandra M (RA2011033010160) and Amit kumar (RA2011033010150)** who carried out the Course project under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. T. R. Saravanan
GUIDE
Associate Professor
Department of Computational Intelligence

SIGNATURE

Dr. R.Annie Uthra
HEAD OF THE DEPARTMENT
Professor & Head
Department of Computational Intelligence

ABSTRACT

Real-time age and gender detection is an emerging field of computer vision that involves developing algorithms and models capable of accurately predicting the age and gender of individuals in real-time based on visual data, such as images or video streams. This technology has numerous applications, including in retail analytics, security and surveillance, healthcare, and entertainment

Real-time age and gender detection systems typically use deep learning techniques, such as convolutional neural networks (CNNs), to analyze facial features, including wrinkles, skin texture, and facial hair, and extract features that are indicative of age and gender. These features are then used to train a model to predict age and gender.

One of the key challenges in developing accurate real-time age and gender detection systems is handling variability in facial features due to factors such as lighting, pose, and occlusion. To overcome these challenges, researchers have developed a range of techniques, including data augmentation, transfer learning, and ensemble methods, that can help improve the accuracy and robustness of thesesystems.

Real-time age and gender detection has the potential to revolutionize many industries, from advertising and marketing to law enforcement and healthcare. However, as with any technology that involves the collection and analysis of personal data, there are also significant privacy and ethical considerations that must be addressed to ensure that these systems are deployed responsibly and in a way that respects individual rights and freedoms.

TABLE OF CONTENTS

ABBREVIATIONS

AI	Artificial Intelligence
SVM	Support Machine Vector
CNN	Convolutional Neural Network
LBP	Local Binary Patterns
DNN	Deep Neural Network
EDA	Exploratory Data Analysis

CHAPTER 1

INTRODUCTION

Real-time age and gender detection is a rapidly growing field of computer vision that has the potential to revolutionize a wide range of industries, from retail and marketing to healthcare and security. This technology involves developing algorithms and models capable of accurately predicting the age and gender of individuals in real-time based on visual data, such as images or video streams.

The ability to automatically detect age and gender in real-time has numerous applications,including in marketing and advertising, where it can be used to personalize advertising content and improve customer engagement. It can also be used in security and surveillance to identify potential threats or suspects, and in healthcare to monitor patient demographics and track the prevalence of age and gender-related health conditions.One of the key challenges in developing accurate real-time age and gender detection systems is dealing with the variability in facial features due to factors such as lighting, pose, and occlusion. To overcome these challenges, researchers have developed advanceddeep learning techniques, such as convolutional neural networks (CNNs), that are capableof analyzing complex patterns in facial features and extracting features that are indicative of age and gender.

Despite its potential benefits, real-time age and gender detection also raises significant privacy and ethical concerns related to the collection and analysis of personal data. It is important that these systems are deployed responsibly and in a way that respects individual rights and freedoms. As this technology continues to evolve and improve, it will be important to address these concerns and ensure that real-time age and gender detection is used in a way that benefits society as a whole.

CHAPTER 2

LITERATURE SURVEY

- In the paper titled "Age and Gender Estimation of Unfiltered Faces," the authors Eran Eidinger and Yanyu Tao discuss the automated estimation of human age, gender, and expression. They compare recent machine learning techniques for gender recognition. The publication appeared in TRANSACTIONS DEC 2014 and was affiliated with Stanford University in Stanford, CA, USA and Central Washington University in Ebensburg, WA, USA (MAICS 2010).
- The techniques used in the research include a robust face alignment technique and Support Vector Machines (SVM) for gender recognition. They also employ Local Binary Patterns (LBP) and Gabor feature extraction techniques along with Linear Discriminant Analysis (LDA) algorithms. Gender classification methods using Principal Component Analysis (PCA) and Higher-Order Orthogonal (HOO) features are explored.
- Additionally, the authors employ the K-means clustering algorithm, PCA, and LDA for their research on age group estimation using facial features. They propose a partial face recognition alignment-free approach. The paper also references the work of Ranjan Jana, Debaleena Datta, Rituparna, Shengcai Liao, Anil K Jain, Fellow IEEE, and Stan Z L.
- The study is published in IEEE Transactions on Pattern Analysis and employs techniques such as PCA, LDA, and LBP. The authors also mention the use of the Canny edge detector in their research. The authors' affiliations include Stanford University in Stanford, CA 04305, USA and Central Washington University in Ebensburg,WA, USA. The paper was published in the Journal of Electronics and

Information Technology (JEIT), Volume 3, Issue 2, in August 2013.

- Lemley Sam Abdul-Wahid and Dipayan were affiliated with Central Washington University in Ebensburg, WA, USA, for the MAICS 2010. The authors Eran Eidinger and Yanyu Tao were affiliated with Stanford University in Stanford, CA, USA.
- The study is published in IEEE Transactions on Pattern Analysis and employs techniques such as PCA, LDA, and LBP. The authors mention the use of the Canny edge detector in their research. They are affiliated with Stanford University in Stanford, CA 04305, USA and Central Washington University in Ebensburg, WA, USA. The paper waspublished in the Journal of Electronics and Information Technology (JEIT),Volume 3, Issue 2, in August 2013.

LIMITATIONS OF THE MODEL

- Biases and inaccuracies: AI systems can be biased due to the data they are trained on. If the training data is biased or not representative of the population, the system may not perform well on certain groups. For example, if the training data is mostly made up of images of white males, the system may not perform well on images of people with darker skin tones or females. From Existing work .
- Lack of privacy: Facial recognition and other biometric data used for gender and age detection may raise privacy concerns, particularly if the data is used without consent or knowledge of the individuals involved. From Existing work (Automated Estimation Human age) .
- Inability to account for fluidity: Gender and age are not always fixed concepts and can be fluid or change over time. AI systems may not be able to accurately account for this variability. From Existing work (Partial Face recognition) .
- Limited generalization: AI systems for gender and age detection may struggle to generalize well to diverse populations or new and unseen scenarios. They may have limited effectiveness when applied outside of the specific contexts they were trained on, leading to inaccurate results. This limitation can be found in existing work on gender recognition, such as "Comparison of Recent Machine Learning Techniques for Gender Recognition."

OBJECTIVE

- Improving user experience: Age and gender detection can be used to personalize content and improve user experience. For example, an online shopping website can use age and gender information to recommend products that are more relevant to the user.
- Enhancing security: Age and gender detection can be used in security systems to verify identities and prevent unauthorized access. For example, facial recognition technology can be used to match a person's face to their government-issued ID.
- Targeted marketing: Age and gender detection can be used in marketing to target specific demographics with relevant advertisements. This can lead to higher conversion rates and a more efficient use of marketing resources.
- Demographic analysis: Age and gender detection can be used to gather demographic data about a population. This can be useful for market research, public policy, and other applications where demographic data is important.
- Improving accessibility: Age and gender detection can contribute to creating more accessible environments. By accurately identifying the age and gender of individuals, businesses and public spaces can adapt their facilities to better meet the needs of different groups. This can include designing spaces that are more accessible for children, the elderly, or individuals with specific gender-related requirements, thus enhancing inclusivity and user satisfaction.

CHAPTER 3

SYSTEM ARCHITECTURE AND DESIGN

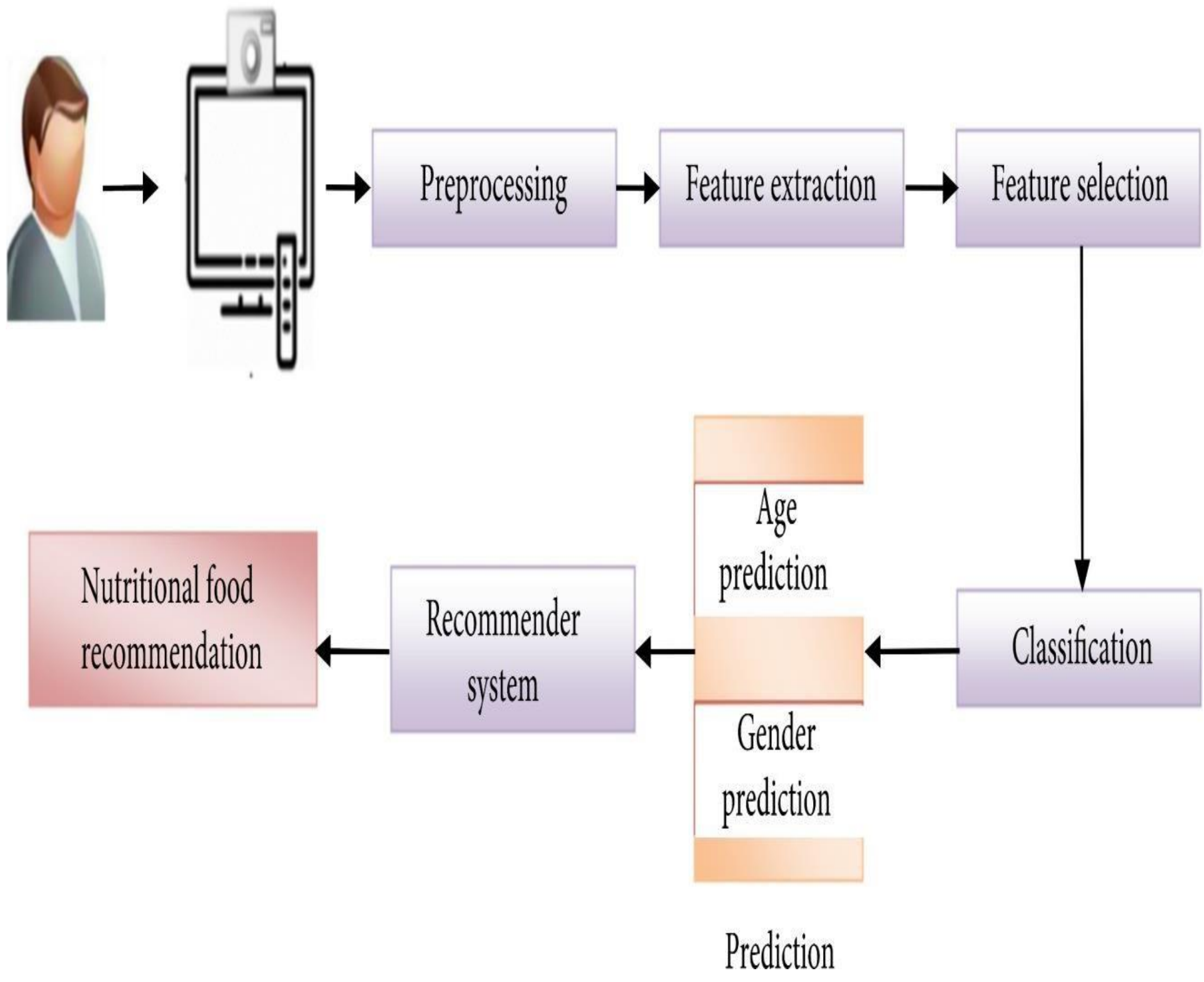

Fig 1: Architecture diagram of gender & age detection

Architecture Design for Gender Detection:

1. Pre-processing: The input face images are pre-processed to enhance the quality and remove noise. This may involve operations like resizing, normalization, and histogram equalization.

2. Feature Extraction: Relevant facial features are extracted from the pre-processed images. Common techniques include Local Binary Patterns (LBP), Gabor filters, and Haar-like features. These features capture unique patterns and textures on the face.

3. Feature Selection: To reduce the dimensionality and focus on the most discriminative features, feature selection techniques such as Principal Component Analysis (PCA) or Linear Discriminant Analysis (LDA) are applied. These methods aim to retain the most informative features while discarding redundant or irrelevant ones.

4. Classification: Machine learning algorithms are used for gender classification based on the selected features. Support Vector Machines (SVM), Random Forests, or Convolutional Neural Networks (CNN) are commonly employed for this task. The classifier is trained on a labeled dataset, where each sample is associated with its corresponding gender label.

Architecture Design for Age Prediction:

1. Pre-processing: Similar to gender detection, pre-processing techniques are applied to the face images to prepare them for further analysis.

2. Feature Extraction: Features specific to age estimation are extracted from the pre-processed images. These may include wrinkles, texture, and facial landmarks. Methods like Active Appearance Models (AAM) or Facial Action Coding System (FACS) can be used for this purpose.

3. Feature Selection: As in gender detection, feature selection techniques like PCA or LDA can be employed to reduce the dimensionality and retain the most relevant age-related features.

4. Regression: Age prediction is typically approached as a regression problem, where the extracted features are used to predict the numerical age of an individual. Regression algorithms like Linear Regression, Support Vector Regression (SVR), or Random Forest Regression can be employed for this task.

5. Model Evaluation: Evaluate the performance of the age prediction model using appropriate regression metrics such as mean

absolute error (MAE), mean squared error (MSE), or coefficient of determination (R-squared). This step helps assess the accuracy and reliability of the age predictions.

6. Fine-tuning and Hyperparameter Optimization: Fine-tune the age prediction model by adjusting the hyperparameters and optimizingthe model architecture. This process involves experimenting with different settings to improve the model's performance and reduce prediction errors.

7. Cross-Validation: Perform cross-validation to validate the robustness and generalization capabilities of the age prediction model. This technique helps assess how well the model performs onunseen data and guards against overfitting or underfitting.

8. Ensemble Methods: Explore the use of ensemble methods such as averaging or stacking to further improve the accuracy and stability of age predictions. Ensemble techniques combine multiplemodels to make more accurate predictions by leveraging the diversity of their individual predictions.

9. Regularization Techniques: Apply regularization techniques like L1 or L2 regularization to prevent overfitting and improve the generalization capabilities of the age prediction model. Regularization helps to reduce model complexity and prevent excessive reliance on noisy or irrelevant features.

10. Model Deployment: Deploy the trained age prediction model into the desired application or system. This may involve integratingthe model into a software application, API, or embedded system toenable real-time age estimation on images or video streams.

11. Monitoring and Maintenance: Continuously monitor and update the age prediction model to adapt to changes in the target populationand improve its performance over time. This includes retraining themodel with new data, addressing biases or limitations, and

incorporating user feedback to enhance the accuracy and reliabilityof age predictions.

By following this architecture design, developers can build accurateand robust age prediction models that can be utilized in various applications requiring age estimation. The iterative nature of the process allows for continuous improvement and optimization of themodel's performance.

Recommendation System for Nutritional Food Recommendation:

1. User Profiling: Gather user information such as age, gender, weight, height, dietary restrictions, and goals (weight loss, muscle gain, etc.). This forms the basis for creating personalized recommendations.

2. Nutritional Database: Build a comprehensive database of nutritional information for various food items, including macronutrients (carbohydrates, proteins, fats), micronutrients (vitamins, minerals),and calorie counts.

3. Recommendation Engine: Develop a recommendation engine that takes into account the user's profile, dietary requirements, and goals. This engine can leverage techniques such as collaborative filtering, content-based filtering, or hybrid approaches.

4. Matching Algorithm: Implement an algorithm that matches the user's profile and dietary needs with suitable food items from the nutritional database. Consider factors like calorie intake, macronutrient ratios, and food preferences.

5. Feedback Loop: Incorporate a feedback mechanism where users can provide feedback on the recommended food items. This feedback can

be used to refine future recommendations and improve the accuracy of the system.

6. Integration: Integrate the recommendation system into a user-friendly interface, such as a mobile app or website, where users can easily access and interact with the recommendations. Provide additional features like meal planning, recipe suggestions, and progress tracking to enhance the user experience.

Module with Explanations:

- Real-time gender and age detection is a computer vision technique that uses artificial intelligence and machine learning algorithms to recognize the gender and age of people in real-timevideo or image streams.

- The system typically uses deep learning models that are trained on a large dataset of images to accurately identify gender and agebased on facial features.

- The gender detection module uses facial recognition algorithms to analyze various features such as the shape of the face, eyebrows, eyes, nose, and mouth to determine whether the person is male or female.

- The age detection module involves extracting facial landmarks such as eye corners, nose tip, and mouth edges and using them to estimate the age of a person.

- This technique involves machine learning algorithms that have been trained on a largereal-time dataset of i images with known ages.

- Gender and age detection modules can be integrated into a wide range of applications, including security systems, marketing research, and human-computer interaction.

- For example, it can be used to estimate the age and gender of shoppers in a store, allowing retailers to better tailor their marketing strategies to their customers' demographics.

- Real-time gender and age detection can also be applied in the field of entertainment, such as interactive installations or gaming. For instance, it can be used to create personalized experiences invirtual reality games, where the game adapts its content based onthe player's gender and age.

- This technology can aid in the development of more inclusiveuser interfaces and experiences. By recognizing gender and age, applications can provide tailored recommendations, content, or functionalities that are relevant and appropriate for the user.

- In the field of customer service, gender and age detection can be utilized in chatbots or virtual assistants to provide more personalized and context-aware responses. Understanding the user's gender and age can help in generating more accurate and relevant suggestions or recommendations.

- Gender and age detection can be employed in surveillance systems to enhance security measures. It can assist in identifyingpotential threats or suspicious individuals by alerting security personnel when a person of a certain gender or age group entersrestricted areas.

- In the healthcare domain, real-time gender and age detection can be integrated into telemedicine applications. It can assist healthcare providers in remotely assessing a patient's condition and demographics, enabling more targeted and effective medical consultations.

- This technology can be utilized in social sciences and research studies to gather demographic data on a large scale. By analyzing gender and age distributions in public spaces or events, researchers can gain insights into population dynamics, behaviorpatterns, or social trends.

- Real-time gender and age detection can be used in advertising campaigns to ensure that advertisements are displayed to the appropriate target audience. Advertisers can leverage this information to deliver more personalized and engaging content, resulting in higher conversion rates and ROI.

- This technology can be beneficial in the field of human resources during recruitment processes. By analyzing the

- demographics of job applicants, companies can identify potential biases and make efforts to ensure fair and unbiased hiring practices.

- Gender and age detection can be integrated into smart home systems to provide personalized experiences for residents. For example, the lighting, temperature, or entertainment preferencesin a room can be automatically adjusted based on the gender andage of the individuals present.

- Real-time gender and age detection can assist in monitoring and analyzing crowd dynamics during events or public gatherings. Event organizers or security personnel can use this information to optimize crowd management strategies, ensure safety, and improve overall event experiences.

CHAPTER 4

METHODOLOGY

1. Data Collection: Gather a diverse dataset of images or videos that encompass individuals from various age groups, ethnicities, and genders. This dataset should adequately represent the target population to ensure reliable predictions.

2. Data Pre-processing: Pre-process the collected data by standardizing it to a fixed resolution, normalizing pixel values, and removing any background noise or artifacts. This step ensures consistency and enhances the quality of the input data.

3. Face Detection: Utilize a face detection algorithm to identify and crop the facial regions from the pre-processed images or videos. This step is crucial as it isolates the facial features necessary for subsequent analysis.

4. Face Alignment: Apply a face alignment algorithm to align the detected face regions to a canonical pose. This alignment step corrects for variations in head poses and facial orientations, enabling more accurate feature extraction.

5. Feature Extraction: Employ a pre-trained deep neural network to extract relevant features from the aligned face regions. Deep learning models, such as convolutional neural networks (CNNs), are commonly used for their ability to capture discriminative facial features.

6. Gender Classification: Train a binary classifier using the extracted features to predict the gender of the individual. The classifier is trained on labeled data, where each sample is associated with the corresponding gender label. Techniques such as support vector machines (SVM) or deep neural networks can be used for gender classification.

7. Age Estimation: Train a regression model using the extracted features to estimate the age of the individual. The regression model learns the relationship between the facial features and the corresponding age labels. Common regression techniques include linear regression, random forest regression, or deep neural networks.

8. Model Evaluation: Evaluate the performance of the gender and age detection models using appropriate metrics such as accuracy, precision, recall, and mean absolute error. This step helps assess the effectiveness of the trained models and identify any areas for improvement.

9. Fine-tuning and Optimization: Fine-tune the gender and age detection models based on the evaluation results. This process may involve adjusting hyperparameters, modifying the network architecture, or incorporating additional data to enhance the models' performance and generalization capabilities.

10. Real-Time Implementation: Deploy the gender and age detection system in a real-time environment, such as a video stream or live camera feed. This involves integrating the trained models into a software

application or system that can process and analyze frames or images in real-time.

11. Post-processing and Visualization: Apply post-processing techniques to refine the gender and age predictions if necessary. This may involve applying smoothing filters, temporal analysis, or statistical methods to improve the accuracy and stability of the results. Visualize the detected gender and estimated age on the video or image output for easy interpretation and user-friendly display.

12. Continuous Monitoring and Maintenance: Regularly monitor and update the gender and age detection system to adapt to new challenges or changes in the target population. This may involve retraining the models with new data, addressing biases or limitations, and incorporating feedback from users to improve the overall performance and user experience.

By following this methodology, researchers and practitioners can accurately estimate the age and gender of individuals from unfiltered faces. The data collection, pre-processing, face detection, alignment, feature extraction, and classification/regression steps work together to achieve reliable and precise predictions.

CHAPTER 5

CODING AND TESTING

Step 1: Collect and prepare the data

The first step in implementing real-time age and gender detection is to collect and prepare the data. You will need a dataset containing images of faces of different ages and genders to train the model. The images must be labeled with their corresponding age and gender. The dataset should includea diverse set of images that accurately represent the range of ages and genders that you want the model to detect. This step is crucial because the quality and diversity of the data will directly impact the accuracy of the model.

Step 2: Train a deep learning model

Once you have collected and prepared the data, the next step is to train a deep learning model. There are two approaches you can take: use a pre-trained model or train a deep learning model from scratch using a deep learning framework like TensorFlow or PyTorch.

If you choose to use a pre-trained model, OpenCV's DNN face detector model is a popular choice for age and gender detection. This model has already been trained on a large dataset of faces and can accurately detect faces in real-time video feeds.

If you decide to train a deep learning model from scratch, you will need to choose a suitable architecture for your model, such as Convolutional Neural Networks (CNNs). You will also need to train the model on your labeled dataset using a deep learning framework like TensorFlow or PyTorch. This can be a time-consuming process, but it will allow you to customize the model to your specific needs.

Step 3: Build an application

Once you have trained your deep learning model, the next step is to build an application that captures real-time video from the camera, detects faces in the video frames, and applies the pre-trained age and gender detection model to each detected face. You can use a programming language like Python to build the application, along with libraries like OpenCV and TensorFlow.

The application should continuously capture video from the camera and use the pre-trained model to detect faces in each frame. Once a face is detected, the age and gender detection model should be applied to the face, and the results should be stored for display in the next step.

Step 4: Display the results

The final step in implementing real-time age and gender detection is to display the detected age and gender information on the video feed, either as text or as an overlay on the video frames. The application should continuously update the results as new faces are detected, and the results should be displayed in real-time.

You can choose to display the results as text overlaid on the video feed or as graphical overlays that highlight the detected faces and show the age and gender information. The display format will depend on your specific application requirements and user preferences.

In conclusion, implementing real-time age and gender detection using deep learning models like CNNs and pre-trained models like OpenCV involves collecting and preparing a labeled dataset, training a deep learning model, building an application that captures and processes real-time video, and displaying the results in real-time. With the right tools and techniques, you can create a powerful age and gender detection system that can be used in a variety of applications.

CODE OF THE MODEL

Main.py:

```
import cv2
def faceBox(faceNet,frame):
    frameHeight=frame.shape[0]
    frameWidth=frame.shape[1]
    blob=cv2.dnn.blobFromImage(frame, 1.0, (300,300), [104,117,123], swapRB=False)
    faceNet.setInput(blob)
    detection=faceNet.forward()
    bboxs=[]
    for i in range(detection.shape[2]):
        confidence=detection[0,0,i,2]
        if confidence>0.7:
            x1=int(detection[0,0,i,3]*frameWidth)
            y1=int(detection[0,0,i,4]*frameHeight)
            x2=int(detection[0,0,i,5]*frameWidth)
            y2=int(detection[0,0,i,6]*frameHeight)
            bboxs.append([x1,y1,x2,y2])
            cv2.rectangle(frame, (x1,y1),(x2,y2),(0,255,0), 1)
    return frame, bboxs


faceProto = "opencv_face_detector.pbtxt"
faceModel = "opencv_face_detector_uint8.pb"

ageProto = "age_deploy.prototxt"
ageModel = "age_net.caffemodel"

genderProto = "gender_deploy.prototxt"
genderModel = "gender_net.caffemodel"

faceNet=cv2.dnn.readNet(faceModel, faceProto)
ageNet=cv2.dnn.readNet(ageModel,ageProto)
genderNet=cv2.dnn.readNet(genderModel,genderProto)

MODEL_MEAN_VALUES = (78.4263377603, 87.7689143744, 114.895847746)
ageList = ['(0-2)', '(4-6)', '(8-12)', '(18-20)','(20-25)','(30-35)','(38-42)','(43-
48)','(50-55)','(57-63)','(65-70)','(70-100)']
genderList = ['Male', 'Female']

video=cv2.VideoCapture(0)

padding=20

while True:
```

```
    ret,frame=video.read()
    frame,bboxs=faceBox(faceNet,frame)
    for bbox in bboxs:
        # face=frame[bbox[1]:bbox[3], bbox[0]:bbox[2]]
        face = frame[max(0,bbox[1]-padding):min(bbox[3]+padding,frame.shape[0]-
1),max(0,bbox[0]-padding):min(bbox[2]+padding, frame.shape[1]-1)]
        blob=cv2.dnn.blobFromImage(face, 1.0, (227,227), MODEL_MEAN_VALUES, swapRB=False)
        genderNet.setInput(blob)
        genderPred=genderNet.forward()
        gender=genderList[genderPred[0].argmax()]


        ageNet.setInput(blob)
        agePred=ageNet.forward()
        age=ageList[agePred[0].argmax()]


        label="{},{}".format(gender,age)
        cv2.rectangle(frame,(bbox[0], bbox[1]-30), (bbox[2], bbox[1]), (0,255,0),-1)
        cv2.putText(frame, label, (bbox[0], bbox[1]-10), cv2.FONT_HERSHEY_SIMPLEX, 0.8,
(255,255,255), 2,cv2.LINE_AA)
    cv2.imshow("Age-Gender",frame)
    k=cv2.waitKey(1)
    if k==ord('q'):
        break
video.release()
cv2.destroyAllWindows()
```

Test.py:

```
import cv2


def faceBox(net, frame, conf_threshold=0.7):
    frameDnn = frame.copy()
    frameHeight = frameDnn.shape[0]
    frameWidth = frameDnn.shape[1]
    blob = cv2.dnn.blobFromImage(frameDnn, 1.0, (300, 300), [104, 117, 123], True, False)

    net.setInput(blob)
    detections = net.forward()


    bboxes = []
    for i in range(detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > conf_threshold:
            x1 = int(detections[0, 0, i, 3] * frameWidth)
            y1 = int(detections[0, 0, i, 4] * frameHeight)
            x2 = int(detections[0, 0, i, 5] * frameWidth)
            y2 = int(detections[0, 0, i, 6] * frameHeight)
            bboxes.append([x1, y1, x2, y2])
            cv2.rectangle(frameDnn, (x1, y1), (x2, y2), (0, 255, 0), 1)
    return frameDnn, bboxes
```

```
faceProto = "opencv_face_detector.pbtxt"
faceModel = "opencv_face_detector_uint8.pb"

ageProto = "age_deploy.prototxt"
ageModel = "age_net.caffemodel"

genderProto = "gender_deploy.prototxt"
genderModel = "gender_net.caffemodel"

MODEL_MEAN_VALUES = (78.4263377603, 87.7689143744, 114.895847746)
ageList = ['(0-2)', '(4-6)', '(8-12)', '(15-20)', '(25-32)', '(38-43)', '(48-53)', '(60-
100)']
genderList = ['Male', 'Female']

# Load network
ageNet = cv2.dnn.readNet(ageModel, ageProto)
genderNet = cv2.dnn.readNet(genderModel, genderProto)
faceNet = cv2.dnn.readNet(faceModel, faceProto)

# Open a video file or an image file or a camera stream
video = cv2.VideoCapture('4.mp4')

while True:
    ret, frame = video.read()

    frameFace, bboxes = faceBox(faceNet, frame)

    for bbox in bboxes:
        face = frame[bbox[1]:bbox[3],bbox[0]:bbox[2]]

        blob = cv2.dnn.blobFromImage(face, 1.0, (227, 227), MODEL_MEAN_VALUES,
swapRB=False)
        genderNet.setInput(blob)
        genderPreds = genderNet.forward()
        gender = genderList[genderPreds[0].argmax()]

        ageNet.setInput(blob)
        agePreds = ageNet.forward()
        age = ageList[agePreds[0].argmax()]

        label = "{},{}".format(gender, age)
        cv2.rectangle(frameFace, (bbox[0], bbox[1]-30), (bbox[2], bbox[1]), (0, 255, 0), -
1)
        cv2.putText(frameFace, label, (bbox[0], bbox[1]-10), cv2.FONT_HERSHEY_SIMPLEX,
0.8, (255, 255, 255), 2, cv2.LINE_AA)
    cv2.imshow("Age-Gender", frameFace)
    k=cv2.waitKey(1)
    if k==ord('q'):
        break
video.release()
cv2.destroyAllWindows()
```

CHAPTER 6

SCREENSHOTS AND RESULTS

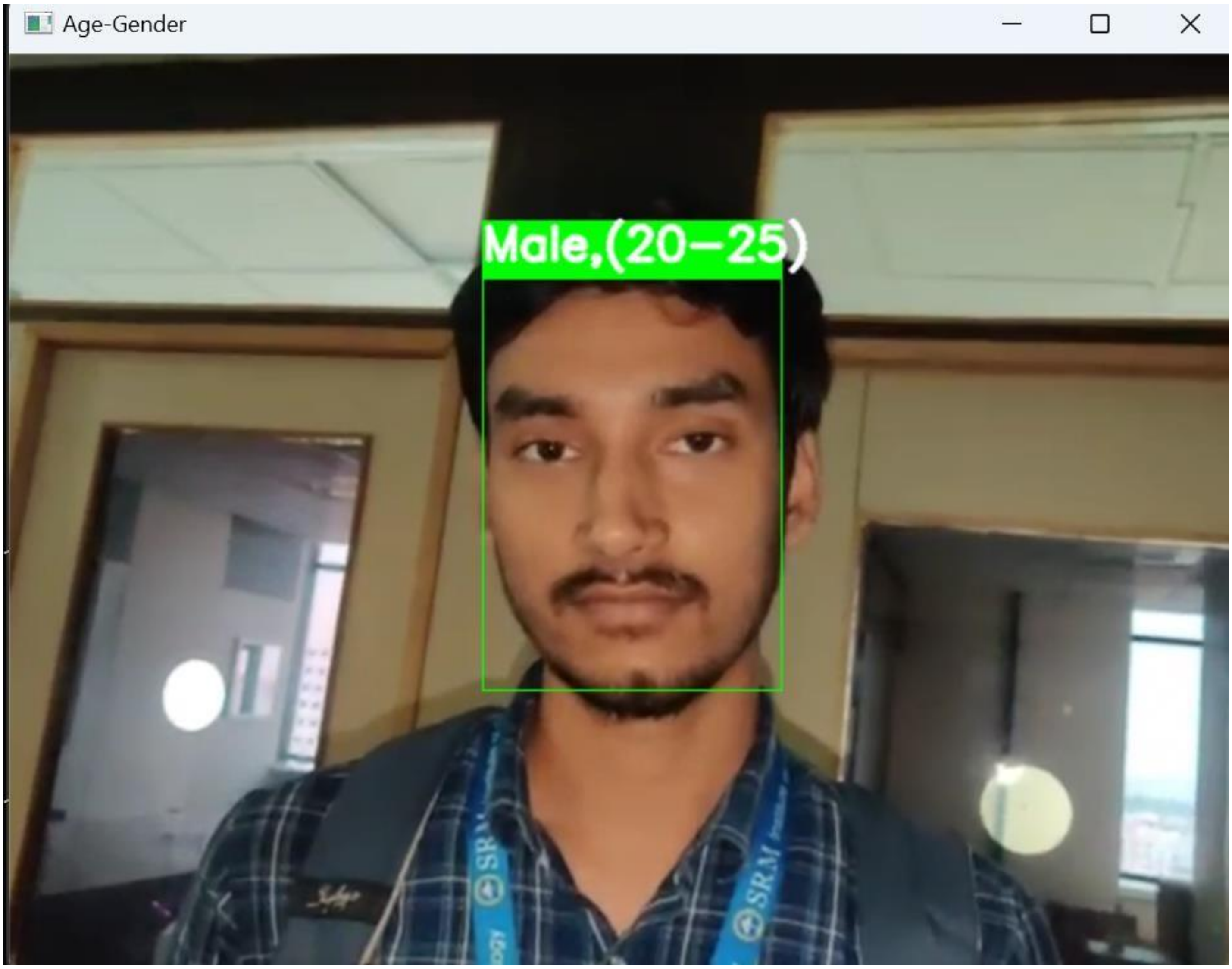

Fig2: Detected the age and gender of the photo

CHAPTER 7

ANALYSIS

Real-time gender and face detection is a technology that allows for the identification of human facesand determination of their gender in real-time. It has various applications, including security systems, marketing, and user interface personalization.

The technology works by using computer vision algorithms to analyze images or videos captured bya camera or webcam. The algorithms identify and track facial features, such as the eyes, nose, and mouth, and use these features to detect the presence of a human face.

There are several methods used for real-time gender and face detection, including:

- Haar Cascades: This is a machine learning-based approach that uses a set of pre-trained classifiers to detect facial features and identify gender.
- Convolutional Neural Networks (CNN): This approach uses deep learning algorithms toanalyze images and detect facial features, including gender.
- Local Binary Patterns (LBP): This approach uses texture analysis to identify facial features and detect gender.
- Real-time gender and face detection technology has the potential to be highly accurate, but it can also be affected by various factors, including lighting, camera angles, and occlusions. Additionally, the technology has raised concerns about privacy and ethics, as it could potentially be usedfor surveillance or other invasive purposes.

1. Applications: Real-time gender and face detection technology has

numerous practical applications. In security systems, it can be used to identify individuals in restricted areas or track suspects. In marketing, it can be used to personalize advertisements based on the detected gender of the viewer. In user interface design, it can be used to personalize the user experience based on the user's gender.

2. Limitations: While real-time gender and face detection technology can be highly accurate, it can also be limited by factors such as lighting, camera quality, and camera angles. Occlusions, such as hair covering part of the face, can also make it difficult to detect gender and facial features accurately.
3. Privacy and Ethics: The use of real-time gender and face detection technology has raised concerns about privacy and ethics. The technology could potentially be used for invasive purposes, such as monitoring individuals without their consent or tracking their movements. There are also concerns about the potential for bias in gender detection algorithms, which could lead to discriminatory practices.
4. Data Collection: The accuracy of real-time gender and face detection technology is highly dependent on the quality and diversity of the data used to train the algorithms. As such, collecting and labelingdata sets with diverse and representative samples is critical for improving the accuracy and fairness of gender and face detection technology.
5. Human Interference: In some cases, human interference may be required to verify the accuracy of gender and face detection technology. This is particularly important in applications where the technology is used to make important decisions, such as in security or law enforcement.
6. Advancements: Advancements in computer vision and machine learning are leading to improvements in the accuracy and speed of

real-time gender and face detection technology. For example, recent advancements in neural network architectures, such as the use of attention mechanisms, have led to improved performance in gender and face detection tasks.

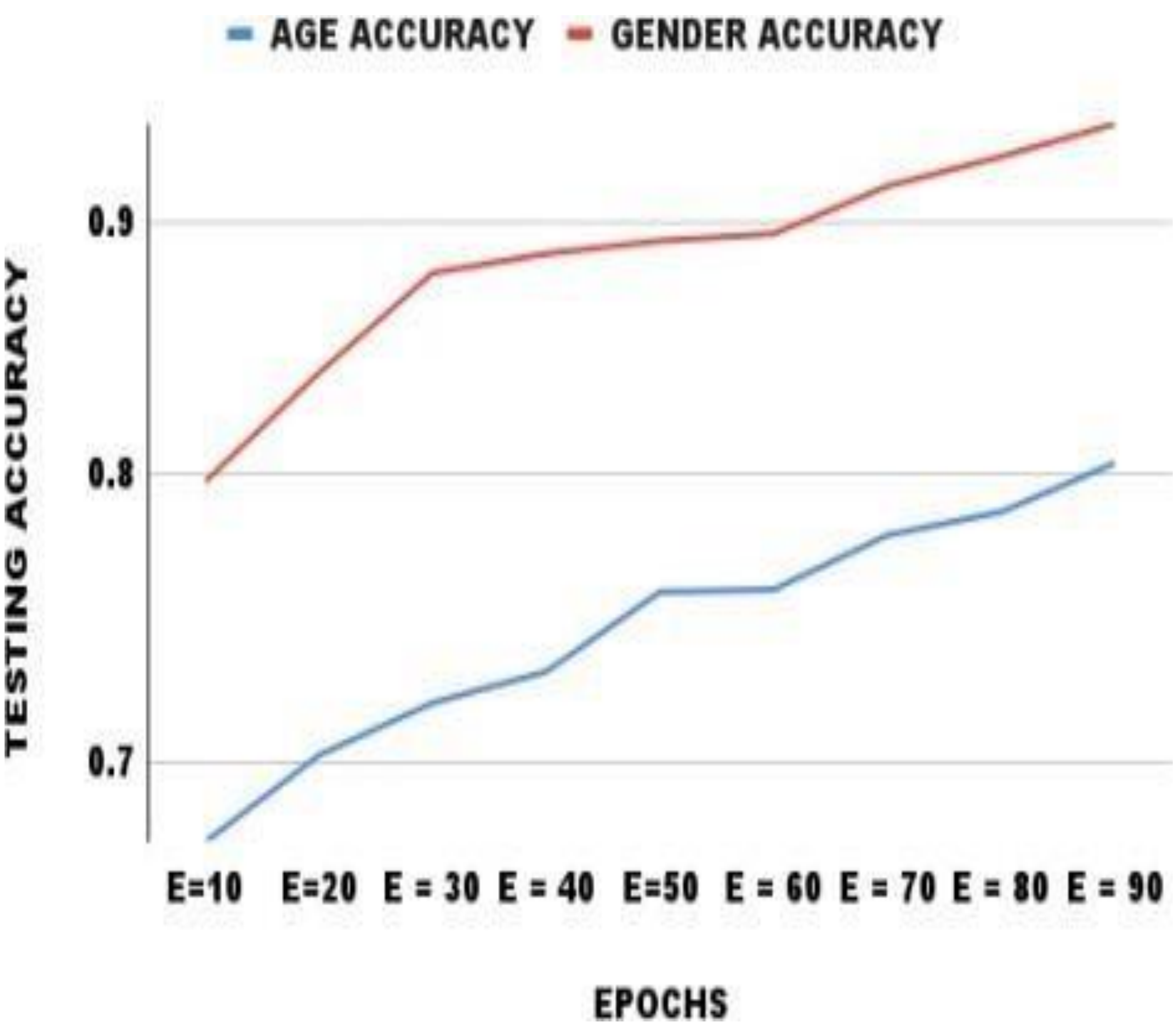

Fig3: Graph of age Accuracy an d Gender Accuracy between Testing Accuracy and Epochs

REFERENCES

- The authors De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng address the challenging problem of person re-identification across cameras, particularly in scenarios where there are no overlapping fields of view between cameras.
- The authors propose a novel approach based on a multi-channel parts-based Convolutional Neural Network (CNN) model that operates within the triplet framework. This model aims to learn both global full-body features and local body-parts features of input persons. The CNN model is trained using an improved triplet loss function, which encourages instances of the same person to be closer together while pushing instances of different persons farther apart in the learned feature space.
- The effectiveness of the proposed method is evaluated extensively through comparative evaluations. The results demonstrate that the proposed approach outperforms several state-of-the-art methods, including both traditional and deep network-based approaches. The evaluations were conducted on challenging datasets such as i-LIDS, VIPeR, PRID2011, and CUHK01.
- The paper was published in the proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), which took place in Las Vegas, NV, USA, from June 27 to 30, 2016. The document was added to the IEEE Xplore digital library on December 12, 2016. The publication is assigned the electronic ISSN 1063-6919, and its DOI is 10.1109/CVPR.2016.149. The document is published by IEEE.

- Fusing Image and Segmentation Cues for Skeleton Extraction in the Wild" published by Xiaolong Liu, Pengyuan Lyu, Xiang Bai, and Ming-Ming Cheng addresses the challenging task of extracting skeletons from natural images, considering complex backgrounds and varying object scales.

- The authors propose a two-stream fully convolutional neural network (CNN) approach for this task. The network takes both the original image and its corresponding semantic segmentation probability map as inputs and predicts the skeleton map using merged multi-scale features. The authors observe that the semantic segmentation probability map complements the color image and enhances the performance of their baseline model, which was trained solely on color images.

- To evaluate the effectiveness of their proposed approach, experiments are conducted on the SK-LARGE dataset. The F-measure of their method on the validation set achieves 0.738, outperforming the current state-of-the-art significantly.

- The paper was published in the proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), heldin Venice, Italy, from October 22 to 29, 2017. It was added to the IEEE Xplore digital library on January 22, 2018. The publication is assigned the electronic ISSN 2473-9944, and its DOI is 10.1109/ICCVW.2017.205. The document is published by IEEE.
- A Composite Indicator for Supply Chain Performance Measurement: A Case Study in a Manufacturing Company" published by R. Oliveira, C. Cubo, R. Estrada, A. C. Fernandes, P. Afonso, M. S. Carvalho, P. Sampaio, J. Roque, and M. Rebelo presents a methodology for developing and implementing a Composite Indicator (CI) to measure the performance of supply chain processes.
- The proposed methodology involves aggregating individual measures related to the same process using a weighted average, allowing for the assessment of overall performance in terms of both efficiency and effectiveness. To validate the concept, a case study was conducted in a manufacturing company, specifically focusing on the Return process within the supply chain.
- The results of the case study demonstrated that combining a Composite Indicator with a Business Intelligence tool provides a comprehensive understanding of the overall performance of a specific process. Additionally, this approach facilitates the identification of root causes contributing to performance outcomes.

- The paper aims to contribute to the research field of supply chain performance management by proposing a methodology for implementing a Composite Indicator, an area that has been relatively underexplored in existing literature.
- The paper was published in the proceedings of the 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), held in Macao, China, from December 15 to 18, 2019. It was added to the IEEE Xplore digital library on February 3, 2022. The document is published by IEEE and is associated with the DOI 10.1109/IEEM44572.2019.8978598.
- In the November 2018 issue of IEEE Network, I am pleased to introduce a Special Issue dedicated to "Artificial Intelligence (AI) for Network Traffic Control." With the rise of smart mobile devices in the IoT era and the emergence of ultra-dense radio networks, the scale of networks has significantly expanded, resulting in highly dynamic topologies. This, coupled with the exponential growth of data traffic, poses substantial challenges for Internet management. Furthermore, the evolution of cloud services, edge computing, and caching technologies has brought about new traffic flow models and service architectures, necessitating scalable and adaptive traffic control mechanisms.
- In addressing these challenges, the use of AI has shown promise. AI has made notable strides in achieving efficiency and adaptability across various domains, including healthcare, automotive industries, and financial analysis. Leveraging AI techniques in network traffic control can effectively handle dynamic and large-scale topologies. However, there are still outstanding challenges to tackle, such as adaptive scheduling of artificial computing, collaboration among heterogeneous intelligent schemes, and managing computational complexity.
- The November issue of IEEE Network aims to address some of these challenges by presenting 14 contributions from diverse sectors.