A Project Report on

AGE AND GENDER DETECTION

)

AGE AND GENDER DETECTION

ABSTRACT

Automatic age and gender classification has become relevant to an increasing number of applications, particularly since the rise of social platforms and social media. Nevertheless, performance of existing methods on real-world images is still significantly lacking, especially when compared to the tremendous leaps in performance recently reported for the related task of face recognition. In this document weshow that by learning representations through the use of deep-convolutional neural networks (CNN), a significant increase in performance can be obtained onthese tasks. To this end, we propose a simple convolutional net architecture that can be used even whenthe amount of learning data is limited. We evaluate our method on the recent Audience benchmark for age and gender estimation and show it to dramatically outperform current state-of-the-artmethods.

TABLE OF CONTENTS

Chapter 1

Introduction

Age and gender play fundamental roles in social interactions. Language's reserve different salutations and grammar rules for men or women, and very often different vocabularies are usedwhenaddressingelderscomparedtoyoungpeople.Despitethebasicrolestheseattributes play in our day-to-daylives, the ability to automatically estimate them accurately and reliably from face images is still far from meeting the needs of commercial applications. This is particularly perplexing when considering recent claims to super-human capabilities in the related task of face recognition. Past approaches to estimating or classifying these attributes from face images have relied on differences in facial feature dimensions or "tailored" face descriptors. Most have employedclassificationschemesdesignedparticularlyforageorgenderestimationtasks,including andothers.Fewofthesepastmethodsweredesignedtohandlethemanychallengesofunconstrained imaging conditions. Moreover, the machine learning methods employed by these systems did not fullyexploitthemassivenumbersofimageexamplesanddataavailablethroughtheInternetinorder toimproveclassificationcapabilities.Inthispaperweattempttoclosethegapbetweenautomaticface recognitioncapabilitiesandthoseofageandgenderestimationmethods.Tothisend,wefollowthe successful example laid down by recent face recognition systems: Face recognition techniques describedinthelastfewyearshaveshownthattremendousprogresscanbemadebytheuseofdeep convolutional neural networks (CNN). We demonstrate similar gains with a simple network architecture, designed by considering the rather limited availability of accurate age and gender labelsin existingfacedatasets.WetestournetworkonthenewlyreleasedAdiencebenchmarkfor age and gender classificationof unfiltered face images. We show that despite the very challenging nature of the images in the Adience set and the simplicity of our network design, our method outperforms existing state of the art by substantial margins. Although these results provide a remarkable baseline for deep-learning-based approaches, they leave room for improvements by moreelaboratesystemdesigns,suggestingthattheproblemofaccuratelyestimating ageandgender in the unconstrained settings, as reflected by the Adience images, remains unsolved. In order to provide a foothold for the development of more effective future methods, we make our trained models and classification system publiclyavailable.

1.1 Objective

Age and gender play fundamental roles in social interactions. Language's reserve different salutations and grammar rules for men or women, and very often different vocabularies areusedwhenaddressingelderscomparedto youngpeople.Inthispaperweshowthatbylearning representationsthroughtheuseofdeep-convolutionalneuralnetworks(CNN),asignificantincrease in performance can be obtained on these tasks. To this end, we propose a simple convolutional net architecture that can be used even when the amount of learning data islimited.

We evaluate our method on the recent Adience benchmark for age and gender estimationand show it to dramatically outperform current state-of-the-art methods. Like other branches of facial analysis, automatic aging and gender classification are hindered by a host of factorsincludingilluminationvariation,facialexpressions,andposevariationtomentionbutafew.
Several approaches have been documented in the literature to circumvent theseproblems.

Researchonfacialagingcanbecategorizedintoageestimation,ageprogression,andage invariantfacerecognition(AIFR).Ageestimationreferstotheautomaticlabelingofagegroupsor the specific ages of individuals using information obtained from their faces. Age progression reconstructs the facial appearance with natural aging effects, and AIFR focuses on the ability to identify or verify people's facesautomatically, despite the effects of aging. In this work, we are focused on ageestimation.

Genderclassificationautomaticallyassignsoneofthetwogenderlabels(male/female)to afacialimage.Studieshaveshownthatwehumansareabletodifferentiatebetweenadultmaleand femalefaces with up to 95% accuracy. However, the accuracy rate reduces to just above chance when considering childfaces.

1.2 LiteratureSurvey

Face description with local binary patterns: Application to face recognition.

This paper presents a novel and efficient facial image representation based on local binary pattern (LBP) texture features. The face image is divided into several regions from which the LBP feature distributions are extracted and concatenated into an enhanced feature vector to be used asa facedescriptor.Theperformanceoftheproposedmethodisassessedinthefacerecognitionproblem under different challenges. Other applications and several extensions are alsodiscussed.

Boosting gender identification performance

This paper presents a method based on AdaBoost to identify the sex of a person from a low-resolution grayscale picture of their face. The method described here is implemented in a systemthatwillprocesswellover109images.Thegoalofthisworkistocreateanefficientsystem that is both simple to implement and maintain; the methods described here are extremely fast and have straightforward implementations. We achieve 80% accuracy in sex identification with less than 10 pixel comparisons and 90% accuracy with less than 50 pixel comparisons. The best classifiers published to date use Support Vector Machines; we match their accuracies with as few as 500 comparison operations on a 20× 20 pixel image. The AdaBoost based classifiers presented hereachieveover93%accuracy;thesematchorsurpasstheaccuraciesoftheSVM-basedclassifiers, and yield performance that is 50 timesfaster.

Learning distance functions using equivalence relations

Weaddresstheproblemoflearningdistancemetricsusingside-informationintheform of groupsof"similar"points.WeproposetousetheRCAalgorithm,whichisasimpleandefficient algorithmforlearningafullrankedMahalanobismetric.WefirstshowthatRCAobtainsthesolution to an interesting optimization problem, founded on an information theoretic basis. If the Mahalanobis matrix is allowed to be singular, we show that Fisher's linear discriminant followed by RCA is the optimaldimensionality

reduction algorithm under the same criterion. We then show how this optimization problem is related to the criterion optimized by another recent algorithm for metric learning, which uses the same kind of side information. We empirically demonstrate that learning a distance metric using the RCA algorithm significantly improves clustering performance, similarly to the alternative algorithm. Since the RCA algorithm is much more efficient and cost effective than the alternative, asitonlyusesclosedformexpressionsofthedata,itseemslikeapreferablechoiceforthelearning offullrankMahalanobisdistances.

Facial age estimation based on label-sensitive learning and age-oriented regression.

In this paper, a new age estimation framework considering the intrinsic properties of human ages is proposed, which improves the dimensionality reduction techniques to learn the connectionsbetweenfacialfeaturesandaginglabels.Toenhancetheperformanceofdimensionality reduction,adistancemetricadjustmentstepisintroducedinadvanceto achieveasuitablemetricin the feature space. In addition, to further exploit the ordinal relationship of human ages, the "label-sensitive" concept is proposed, which regards the label similarity during the learning phase of distance metric and dimensionality reduction. Finally, an age-specific local regression algorithmis proposed to capture the complicated aging process for age determination. From the simulation results, the proposed framework achieves the lowest mean absolute error against the existing methods.

Human age estimation with regression on discriminative aging manifold.

Recently, extensive studies on human faces in the human-computer interaction (HCI) fieldreveal significant potentials for designing automatic age estimation systems via face image analysis. The success of such research may bring in many innovative HCI tools used forthe applications of human- centered multimedia communication. Due to the temporal property of age progression, face images with aging features may display some sequential patterns with low-dimensional distributions. In this paper, wedemonstrate that such aging patterns can be effectively extracted from a discriminant subspace learning algorithm and visualized as distinct manifold structures.Throughthemanifoldmethodofanalysisonfaceimages,thedimensionalityredundancy of the original image space can be significantly reduced with subspace learning. A multiple linear regression procedure, especially with a quadratic model function, canbe facilitated by thelow

dimensionality to represent the manifold space embodying the discriminative property. Such a processing has been evaluated by extensive simulations and compared with the state-of-the- art methods. Experimental results on a large size aging database demonstrate the effectiveness and robustness of our proposed framework.

.

Chapter 2

Previous and Related Work

2.1 AgeEstimation

Over the last 15 years, several pieces of research have been published on facial age estimation. The algorithms usually take one of two approaches: age group or age-specific estimation.Theformerclassifiesapersonaseitherchildoradult,whilethelatterismorepreciseas it attempts to estimate the exact age of a person. Each of these approaches can be further decomposed into two key steps: feature extraction and patternlearning/classification.

2.1.1 Featureextraction

Two feature extraction techniques have been used in the literature: local and holistic. The local approach, also known as the part-based or analytic approach, concentrates on salient parts of the face, such as the facial anthropometry and wrinkles. Using local features, the earliest work on age estimation can be traced back to Kwon and Lobo. Two-dimensional (2-D) images were classified into three age groups: babies, young adults, and senior adults. They represented the face as ratios of distancesbetween feature points, as well as using a snakelet transform to represent wrinkles. The ratios were used to discriminate infants from adults, and the snakelets to discriminate young from senior adults. Several other approaches have extended this basic idea, using sobel edge detection with region tagging, Gabor filters and LBP, and Robinson compass masks to define wrinkle and texture features. More detailed craniofacial growth models have also been developed to define the ratios between facial features coupledwith the adaptive retinal sampling method. A drawback of local features is that they are not suited for specific age estimation, because geometric features describe only shape changes that are predominant in childhood and local textures are limited to wrinkles, which manifest in adulthood.

Holistic, also known as global methods, considers the entire face when extracting features. Subspace learning techniques have been used extensively in the literature; these include PCA, neighborhood preserving projections, LPP, orthogonal LPP, locality sensitive discriminant analysis (LSDA), and marginal Fisher analysis (MFA). The AAM a statistical feature extraction method that captures both shape and texture variation, has been the most widely used technique. Lanitis et al. werethe first to perform specific age estimation using the AAMs.

2.1.2 Agelearning

. This has been approached in two main ways: either as a regression problem thereby considering the ordinal relationship between ages or as that of a multiclass classification. Following the latter approach, conventional classification algorithms, such as support vector machines (SVM)and relevance vector machines, have been employed. Estimation via the use of regression was first presented by Lanitis et al. using a quadratic function (QF). Lanitis et al. comparedtheQFtothreetraditionalclassifiers:shortestdistanceclassifiers,multilayerperceptron (MLP), and the Kohonen self-organizing maps. They reported that MLP and QF had the best performance. Geng et al. described aging pattern subspace (AGES), a method that learns aging patternofindividualsandusesAAMforfeatureextraction.Multiplelinearregressionwasproposed by Fu et al. Using Gaussian mixture models, Yan et al. proposedpatch kernel regression. For a comparison of some recent regression algorithms, the reader is referred to the work of Fernández etal.

2.2 Genderclassification

Gender classification is also approached in two major steps: feature extraction and classification. Feature extraction techniques reported in the literature can be categorized into geometricandappearancebased.Geometry-basedmodelsusemeasurementsextractedfromfacial landmarks to describe the face. In one of the earliest works on gender classification, Ferrario et al.used 22 fiducial points to represent the length and width of the face, then Burton et al.deployed 73 fiducial points; afterward discriminant analysis was used by Burton et al. to classify thehuman faces.Inasecondanalysis,theauthorsused30ratiosand30angles.Fellousextendedtheworksof Ferrario et al.and Burtonet al. Out of 40 fiducial points, 22 distances were extracted; these dimensionswerefurtherreducedto5,usingdiscriminantanalysis.Havingexperimentedonasmall DB of 52 faces, the algorithm was reportedto have achieved 95% gender recognition rate. In summary, geometric models maintain only the geometric relationships between facial features, thereby discarding information about facialtexture.

These models are also sensitive to variations in imaging geometry such as pose and alignment.

The classification step is typically achieved using binary classifiers. SVMs have been themost widely used, other classifiers that have been applied include decision trees, neural networks, boosting, bagging, and other ensembles. For more detailed information on gender classification,thereaderisreferredtothereviewbyNgetal.Tosummarizetheliteratureregarding ageestimationandgenderclassification,severalfeatureextractionmethodshavebeenutilizedand adapted by researchers. While the majority of age estimation and gender classificationtechniques havebeendevelopedforgrayscaleimages,techniqueshavealsobeendevelopedforhandlingcolor images. When dealing with color images early researchers treated the three-color channels as independentgrayscaleimages,byconcatenatingthethreechannelsintoasinglelongvector.Under this simple representation, the spatial relationships that exist between the color pixels are destroyed, and the dimension of the image becomes three times that of the classical grayscale model. Furthermore, research has shown that there is high inter channel correlation among the RGBchannelsandthereforesimpleconcatenationresultsinredundancy.Assuch,severalefficient techniquesofincorporatingcolorchannelshavebeensuggested.Thei1i2i3colortransferhasbeen used in the past to decorrelate the RGB channels using Karhunen– Loève transform. Recently, quaternion, a powerful mathematical tool, has been applied to the problem. This has proven to be a good feature extraction method due to its ability to preserve the spatial relationships among R, G,andBchannels.Additionally,itretainstheholisticpropertiesofPCA.Also,quaternionalgebra has been applied to complex-type moments for color images and has been shown to be invariant to image rotation, scale, and translation transformations. However, the method still works inan unsupervised manner, and hence does not take into consideration the class labels of the response variables. ..

Recently,deeplearningconvolutionalneuralnetworks(DLNN),aclassofmachinelearning techniques that perform both automatic supervised and unsupervised feature extraction, as well as transformation for pattern analysis and classification have gained wide popularity among researchers and have been applied directly to the problem of age estimation and gender classification. In general, the methods perform well due to their ability to capture intricate structures in large datasets. Moreover, DLNN eliminates the trouble of hard-engineered feature extraction.

Chapter 3

Proposed System

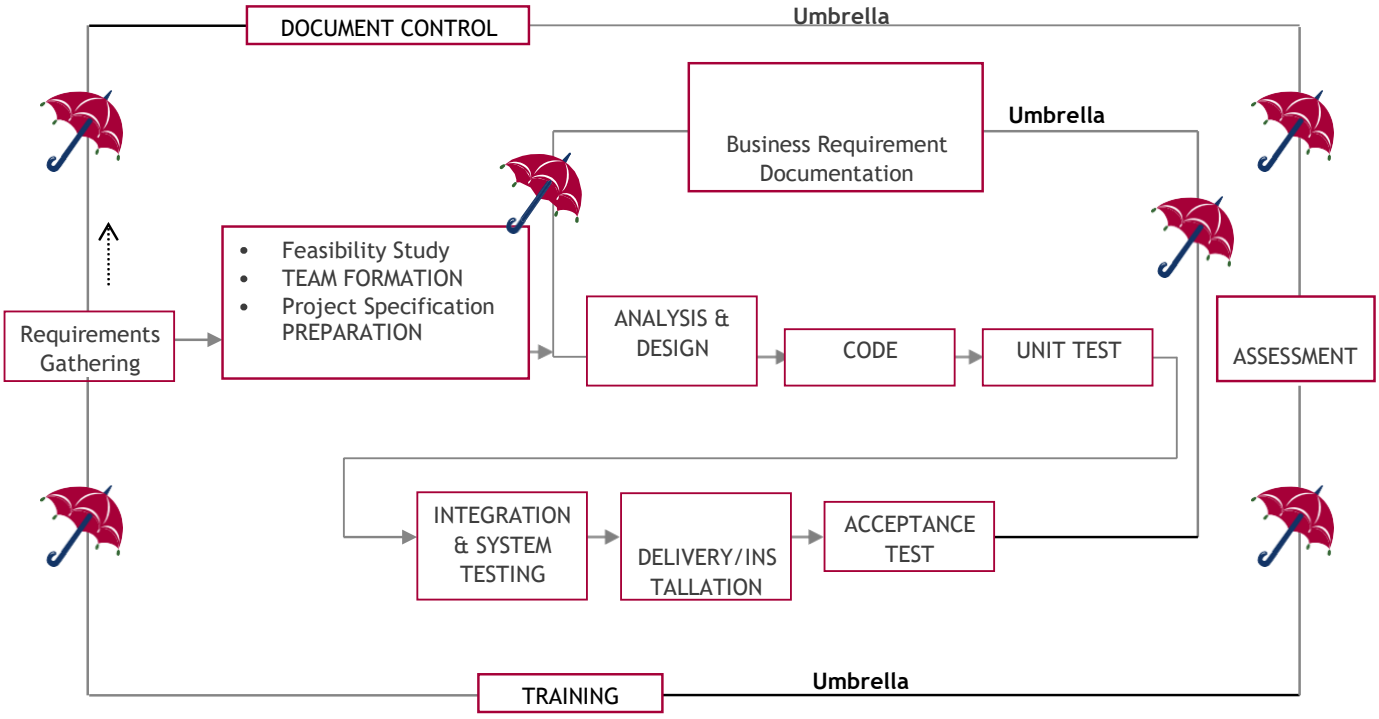
In this paper, we proposed a age and gender predicted framework by using deep-convolutionalneural networks (CNN). Two important conclusions can be made from our results. First, CNN can be usedto provide improved age and gender classification results, evenconsidering the much smaller size of contemporary unconstrained image sets labeled for age and gender. Second, the simplicity of our model implies that more elaborate systems using more training data may well be capable of substantially improving results beyond those reportedhere.

Advantages:

1. Accuracy ishigh.
2. Simplearchitecture

3.1 Process model used withjustification

SDLC (Umbrella model):



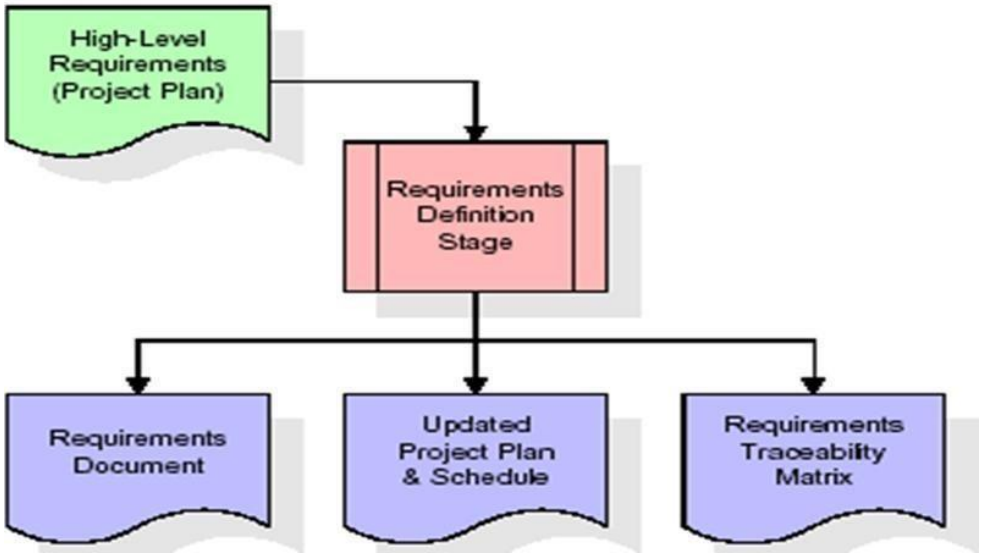
SDLC is nothing but Software Development Life Cycle.

Stages in SDLC:

- RequirementGathering
- Analysis
- Designing
- Coding
- Testing
- Maintenance

Requirements Gathering stage:

The requirements gathering process takes as its input the goals identified in the high-level requirements section of the project plan. Each goal will be refined into a set of one or more requirements.These requirements define the major functions of the intended application, define operational data areas and reference data areas, and define the initial data entities. Major functions include critical processes to be managed, as well as mission critical inputs, outputs and reports. A user class hierarchy is developed andassociated with these major functions, data areas, and data entities. Each of these definitions is termed a Requirement. Requirements are identified by unique requirement identifiers and, at minimum, contain a requirement title and textual description.



These requirements are fully described in the primary deliverables for this stage: the Requirements Document and the Requirements Traceability Matrix (RTM). The requirements document contains complete descriptions of each requirement, including diagrams and references to external documents as necessary. Note that detailed listings of database tables and fields are *not* included in the requirements document.

ThetitleofeachrequirementisalsoplacedintothefirstversionoftheRTM,alongwith the title of each goal from the project plan. The purpose of the RTM is to show that the product components developed during each stage of the software development lifecycle are formally connected to the components developed in priorstages.

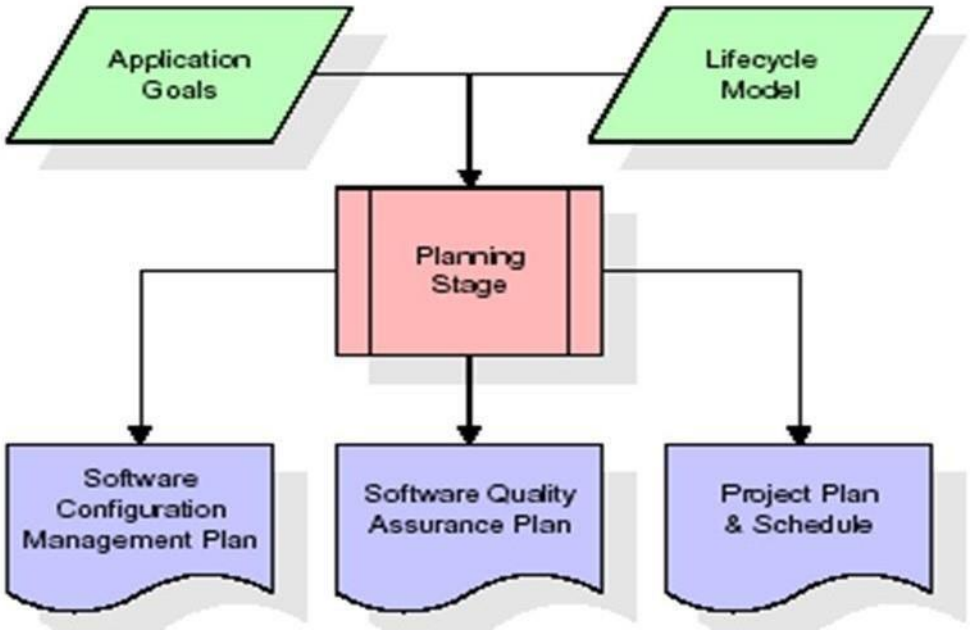
In the requirements stage, the RTM consists of a list of high-level requirements, or goals, by title,with a listing of associated requirements for each goal, listed by requirement title. In this hierarchical listing, the RTM shows that each requirement developed during this stage is formally linkedtoaspecificproductgoal.Inthisformat,eachrequirementcanbetracedtoaspecificproduct goal, hence the term requirementstraceability.

The outputs of the requirements definition stage include the requirements document, the RTM,and an updated project plan.

- Feasibility study is all about identification of problems in aproject.
- No. of staff required to handle a project is represented as Team Formation, in this case onlymodules are individual tasks will be assigned to employees who areworking for thatproject.
- Project Specifications are all about representing of various possible inputs submitting to theserver and corresponding outputs along with reports maintained by administrator.

Analysis Stage:

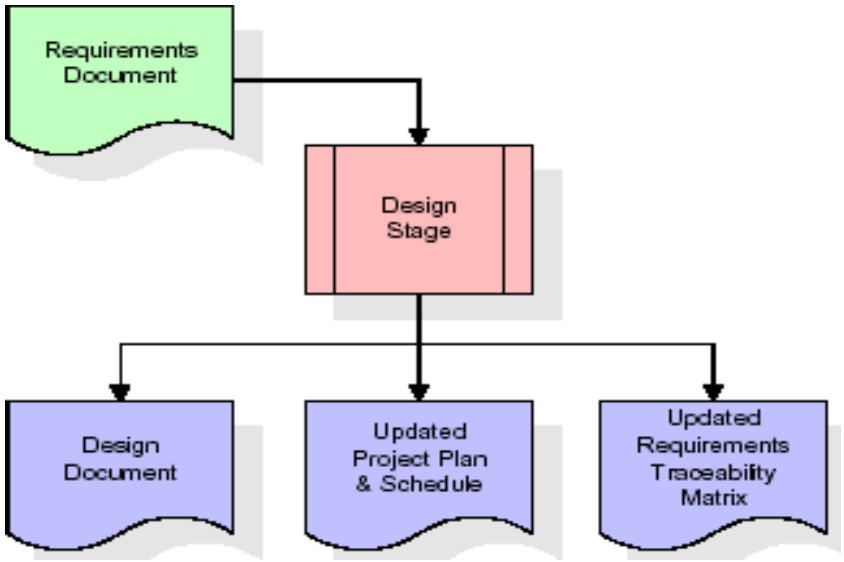
The planning stage establishes a bird's eye view of the intended software product, and uses thistoestablishthebasicprojectstructure,evaluatefeasibilityandrisksassociatedwiththeproject, and describeappropriate management and technicalapproaches.



The most critical section of the project plan is a listing of high-level productrequirements, also referred to as goals. All of the software product requirements to be developed during the requirements definition stage flow from one or more of these goals. The minimum information for each goal consists ofa title and textual description, although additional information and references to external documents may be included. The outputs of the project planning stage are the configuration management plan, the qualityassurance plan, and the project plan and schedule,with a detailed listing of scheduled activities for the upcoming Requirements stage, and high level estimates of effort for the outstages

Designing Stage:

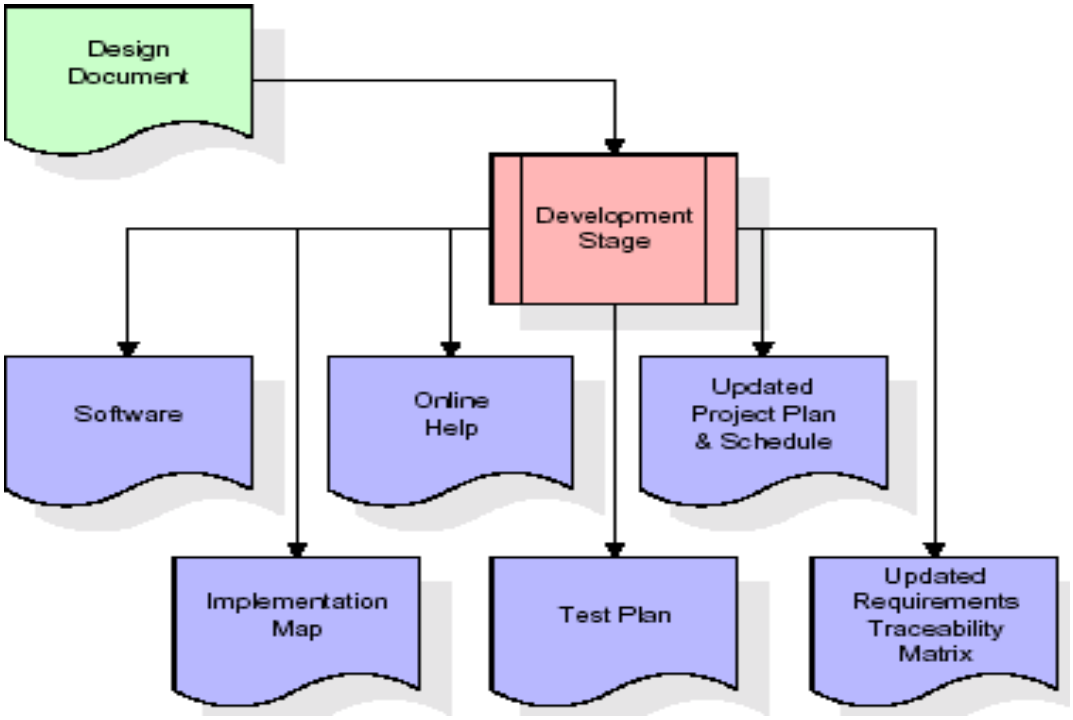
The design stage takes as its initial input the requirements identified in the approved requirements document. For each requirement, a set of one or more design elements will beproducedasaresultofinterviews,workshops,and/orprototypeefforts.Designelementsdescribe the desired software features in detail, and generally include functional hierarchy diagrams, screen layout diagrams, tables of business rules, business process diagrams, pseudo code, and a complete entity-relationship diagram with a full data dictionary. These design elements are intended to describe the software in sufficient detail that skilled programmers may develop the software with minimal additionalinput.



Whenthedesigndocumentisfinalizedandaccepted,theRTMisupdatedtoshowthateach designelement is formally associated with a specific requirement. The outputs of the design stage are the design document, an updated RTM, and an updated projectplan.

Development (Coding) Stage:

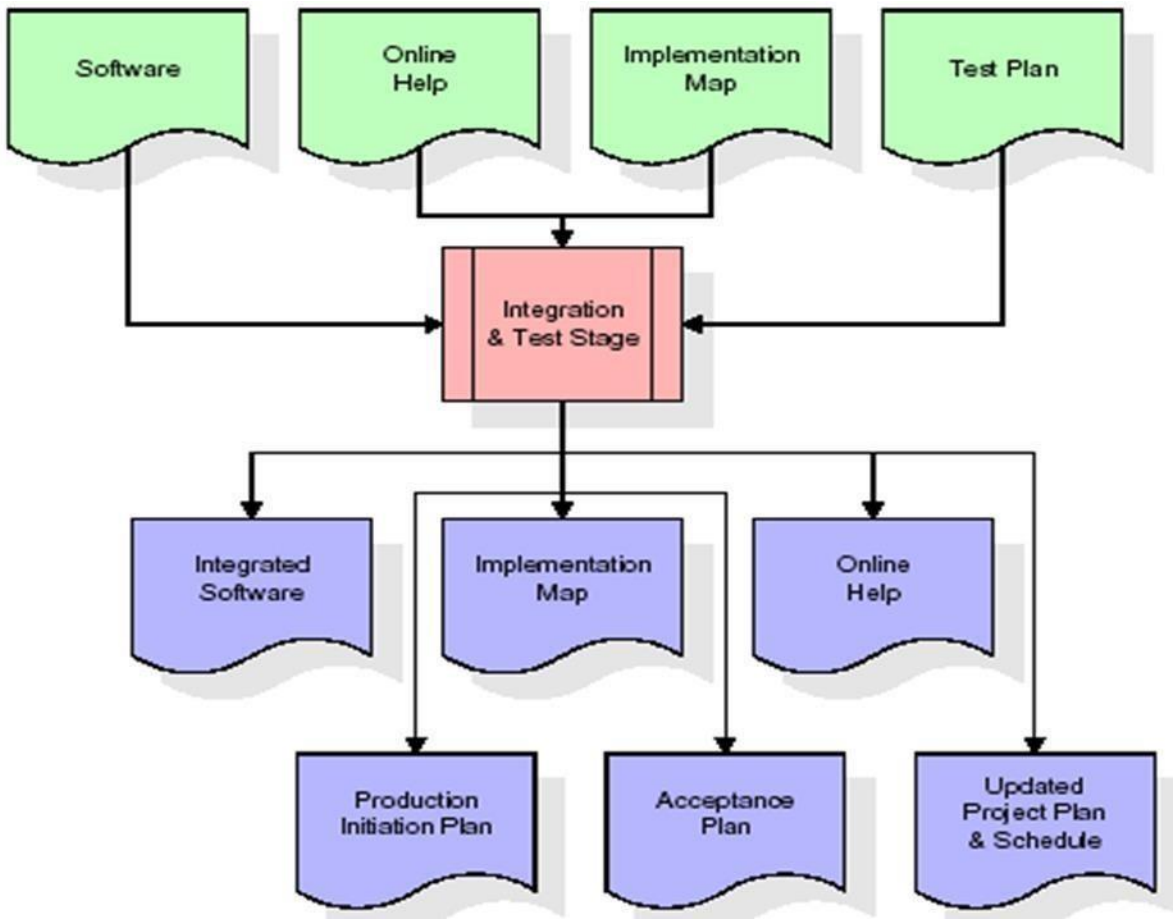
The development stage takes as its primary input the design elements described in theapproveddesigndocument.Foreachdesignelement,asetofoneormoresoftwareartifactswill beproduced.Softwareartifactsincludebutarenotlimitedtomenus,dialogs,anddatamanagement forms,datareportingformats,andspecializedproceduresandfunctions.Appropriatetestcaseswill bedevelopedforeachsetoffunctionallyrelatedsoftwareartifacts,andanonlinehelpsystemwillbe developed to guideusers in their interactions with thesoftware.



TheRTMwillbeupdatedtoshowthateachdevelopedartifactislinkedtoaspecificdesign element, and that each developed artifact has one or more corresponding test case items. At this point, theRTM is in its final configuration. The outputs of the development stage include a fully functional set of software that satisfies the requirements and design elements previously documented, an online help systemthat describes the operation of the software, an implementation map that identifies the primary code entrypoints for all major system functions, a test plan that describes the test cases to be used to validate the correctness and completeness of the software, an updated RTM, and an updated projectplan.

Integration & Test Stage:

During the integration and test stage, the software artifacts, online help, and test data aremigratedfromthedevelopmentenvironmenttoaseparatetestenvironment.Atthispoint,alltest casesareruntoverifythecorrectnessandcompletenessofthesoftware.Successfulexecutionofthe test suite confirms a robust and complete migration capability. During this stage, reference data is finalizedforproductionuseandproductionusersareidentifiedandlinkedtotheirappropriateroles. Thefinalreferencedata(orlinkstoreferencedatasourcefiles)andproductionuserlistarecompiled into the Production InitiationPlan.

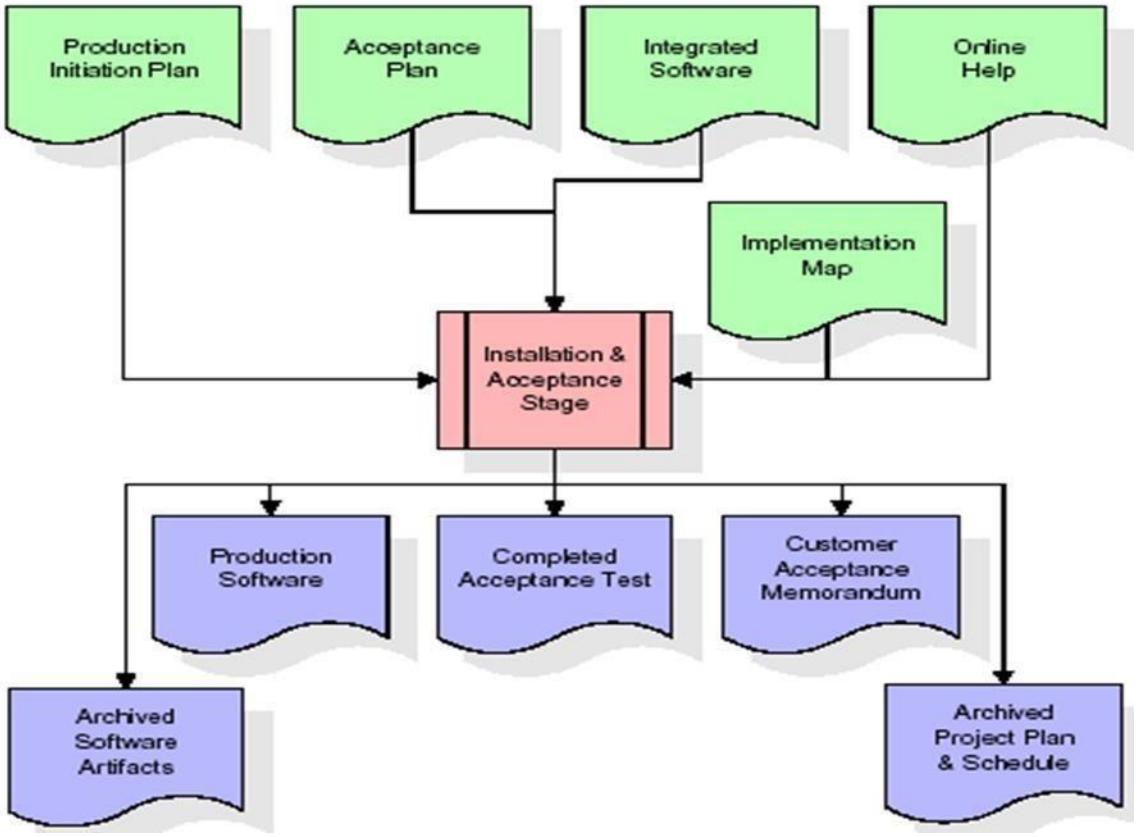


The outputs of the integration and test stage include an integrated set of software, an online help system, an implementation map, a production initiation plan that describes reference data and production users, an acceptance plan which contains the final suite of test cases, and an updated project plan.

- **Installation & AcceptanceTest:**

During the installation and acceptance stage, the software artifacts, online help, and initial production data are loaded onto the production server. At this point, all test cases are run to verify the correctness and completeness of the software. Successful execution of the test suite is a prerequisite to acceptance of the software by the customer.

Aftercustomerpersonnelhaveverifiedthattheinitialproductiondataloadiscorrectandthe testsuitehasbeenexecutedwithsatisfactoryresults,thecustomerformallyacceptsthedeliveryofthe software.



The primary outputs of the installation and acceptance stage include a production application, a completedacceptance test suite, and a memorandum of customer acceptance of the software. Finally,thePDRentersthelastoftheactuallabordataintotheprojectscheduleandlockstheproject as a permanent project record.At this point the PDR "locks" the project by archiving all software items, the implementation map, the source code, and the documentation for future.

3.2 Software requirementspecification

3.2.1 Overalldescription

A Software Requirements Specification (SRS) – a requirements specificationfor a softwaresystemisacompletedescriptionofthebehaviorofasystemtobedeveloped.Itincludesa setofusecasesthatdescribealltheinteractionstheuserswillhavewiththesoftware.Inadditionto use cases, the SRS also contains non-functional requirements. Nonfunctional requirementsare requirements which impose constraints on the design or implementation (such as performanceengineering requirements, quality standards, or designconstraints).

System requirements specification: A structured collection of information that embodies the requirements of a system. A business analyst, sometimes titled system analyst, is responsible for analyzing the business needs of their clients and stakeholders to help identify business problems and propose solutions.Within the systems development lifecycledomain, the BA typically performs a liaison function between thebusiness side of an enterprise and the information technology department or external service providers. Projects are subject to three sorts of requirements:

- Business requirements describe in business terms *what* must be delivered or accomplished to providevalue.
- Product requirements describe properties of a system or product (which could be one of several ways to accomplish a set of businessrequirements.)
- Process requirements describe activities performed by the developing organization. For instance, process requirements could specify Preliminary investigation examine projectfeasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminaryinvestigation:

- **ECONOMICFEASABILITY**

A system can be developed technically and that will be used if installed must still be a goodinvestmentfortheorganization.Intheeconomicalfeasibility,thedevelopmentcostincreating thesystemisevaluatedagainsttheultimatebenefitderivedfromthenewsystems.Financialbenefits mustequalorexceedthecosts.Thesystemiseconomicallyfeasible.Itdoesnotrequireanyaddition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC. There is nominal expenditure and economical feasibility for certain.

- **OPERATIONALFEASIBILITY**

Proposed projects are beneficial only if they can be turned out into information system. Thatwillmeettheorganization'soperatingrequirements.Operationalfeasibilityaspectsoftheproject are to be takenas an important part of the project implementation. This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

- **TECHNICALFEASIBILITY**

Earlier no system existed to cater to the needs of 'Secure Infrastructure Implementation System'.The current system developed is technically feasible. It is a web based user interface for audit workflow atNIC-CSD. Thus it provides an easy access to the users. The database's purpose is to create, establish andmaintain a workflow among various entities in order to facilitate all concernedusersintheirvariouscapacitiesorroles.Permissiontotheuserswouldbegrantedbased on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security.

3.2.2 External Interfacerequirements

User Interface

The user interface of this system is a user friendly python Graphical User Interface.

Hardware Interfaces

The interaction between the user and the console is achieved through python capabilities.

Software Interfaces

The required software is

python.

Operating Environment

Windows XP.

HARDWARE REQUIREMENTS:

- Processor - Pentium–IV
- Speed - 1.1Ghz
- RAM - 256 MB(min)
- HardDisk - 20GB
- KeyBoard - Standard WindowsKeyboard
- Mouse - Two or Three ButtonMouse
- Monitor - SVGA (Super Video GraphicsArray)

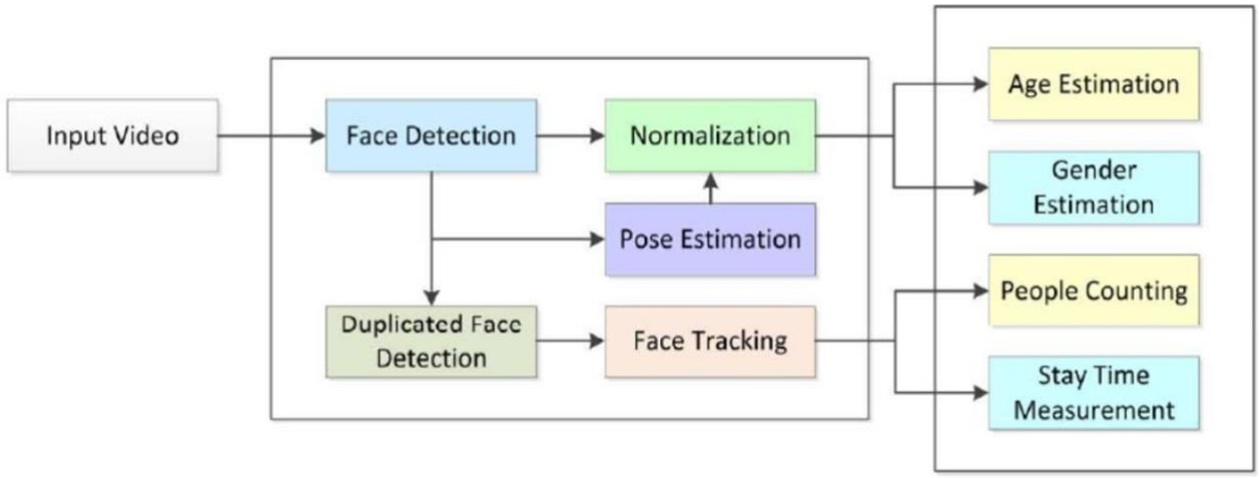
SOFTWARE REQUIREMENTS:

- OperatingSystem - Windows7/8
- ProgrammingLanguage - Python (python3.6.3)

Chapter 4

System Design

4.1 Systemarchitecture



Face recognition and pose estimation:

Inordertoreduceprocessingtime,facedetectionisperformedforonlyeverythirdinput frame,and 64×64 is the minimum size of the face region used for detection. Facial images using localized eyepositions are aligned to apply the face estimation algorithm. In addition, an automatic and monocular head pose estimation method is used for acquiring frontal faces using pitch, yaw and roll values.POSIT (Pose from Orthography and Scaling with ITerations), a hybrid pose estimation algorithm based on both algebraic and optimizing algorithms, is applied for head poseestimation.

POSIT has advantages from both approaches and has good speed and accuracy.

POSIT (Pose from Orthography and Scaling with ITerations), a hybrid pose estimation algorithmbased on both algebraic and optimizing algorithms, is applied for head pose estimation. POSIT has advantages from both approaches and has good speed and accuracy

Thus, POSIT has advantages from both approaches and has good speed and accuracy.

Face tracking and people counting:

The people counting method based on face tracking consists of face detection, face tracking, tracking failure detection, data association and counting modules. The tracking framework is based on amulti-person tracking algorithm (Choi et al., 2015). An HSV (Hue Saturation Value) histogram is employed as an observation model, and Gaussian perturbation is employed as a motion model in a particle filter-based tracking framework. The people counting methodbasedonfacetrackingconsistsoffacedetection,facetracking,trackingfailuredetection, data association and counting modules. The tracking framework is based on a multi-person trackingalgorithm(Choietal.,2015).AnHSV(HueSaturationValue)histogramisemployedas an observation model, and Gaussian perturbation is employed as a motion model in a particle filter-based trackingframework.

Age and gender estimation:

In general, age estimation can be characterized as a multi-class or regression problem,and gender estimation is defined as a two-class problem.

After the face and eyes are detected, the facial image can be normalized as a fixed-size imageusing the localized eye positions. Then, the Gabor-LBP histogram framework (Gao et al., 2008) is usedto extract the features for representation power of the spatial histogram (Lee et al., 2012). UsingGaussian derivative filters, the dimensions of the facial features can be reduced by 60% compared with the Gabor waveletfilters

4.2 Modules

- **Imagepreprocessing:**

Intelligentageandgenderclassifierstackletheclassificationtaskunderunfilteredreal-world settings. Most of those face images are not aligned and non-frontal and also withdifferent degrees ofvariationsinpose,appearance,lighting,andbackgroundconditions.Therefore,thosein-the-wild face images need first to be detected, then aligned, and, finally,used as input for theclassifiers.

- **Facedetection:**

Thefirststageofimagepreprocessingisfacedetection.Thefacedetectionphaselocatesthe faceinaninputimage.Inordertodetectaface,alltheinputimagesarerotated intherangeof−90° to 90° angles and with the step of 5°. After that, the detector selects theinput image with the best outputofthefacedetectorandinacasewherethefaceisnotdetectedinallthe modificationsofthe input image, the original input image is upscaled and face detection algorithm is repeated until a face is detected. The upscaling helps in detecting faces in all the inputimages.

- **Land mark detection and facealignment:**

After face detection, is the facial landmark detection and face alignment phase. Thisimage preprocessing solution is an open-source multi-view facial landmark detection algorithm that uses five landmark detection models, including a frontal model, two half- profile models, and two full profile models.

- **CNNArchitecture:**

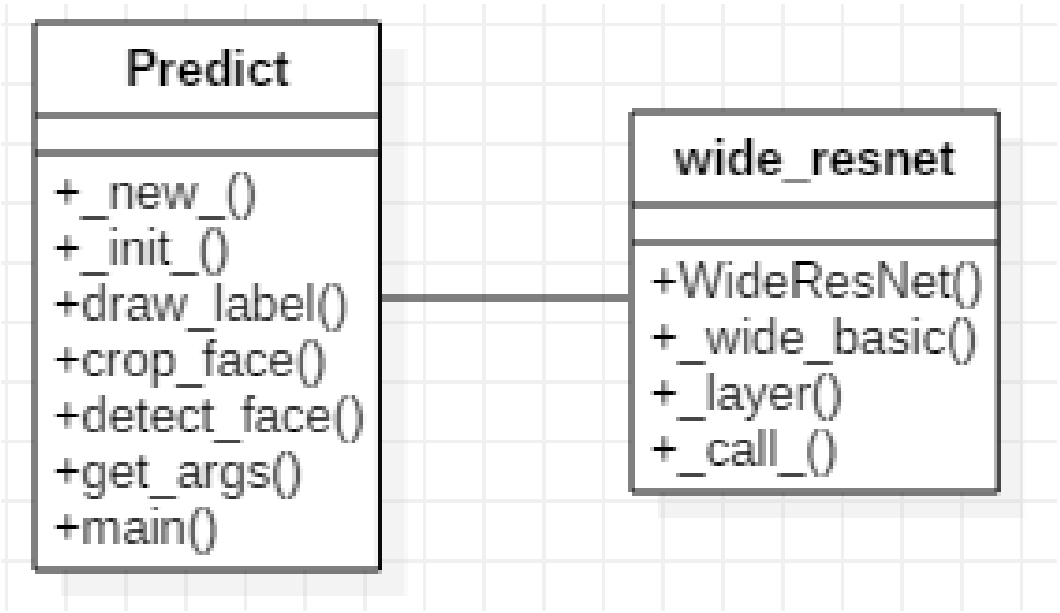
Our CNN architecture is a novel six-layer network, comprising four convolutional and two fully connected layers. The CNN design is an end-to-end sequential deep learningarchitecture, including feature extraction and classification phases. The feature extraction phase has four convolutional layers, with the corresponding parameters, including the number of filters, the kernel size of each filter, and the stride.

4.3 UMLDiagrams

ClassDiagram

The class diagram is the main building block of object-oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the modelsinto programming code. Class diagrams can also be used for data modeling. The classes in a class diagramrepresent both the main objects, interactions in the application and the classes to be programmed. In the diagram, classes are represented with boxes which contain three parts:

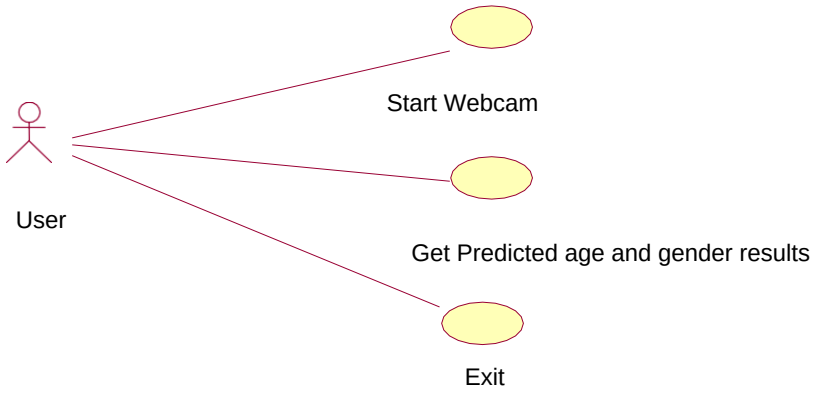
- The upper part holds the name of theclass
- The middle part contains the attributes of theclass
- The bottom part gives the methods or operations the class can take orundertake



Use case Diagram

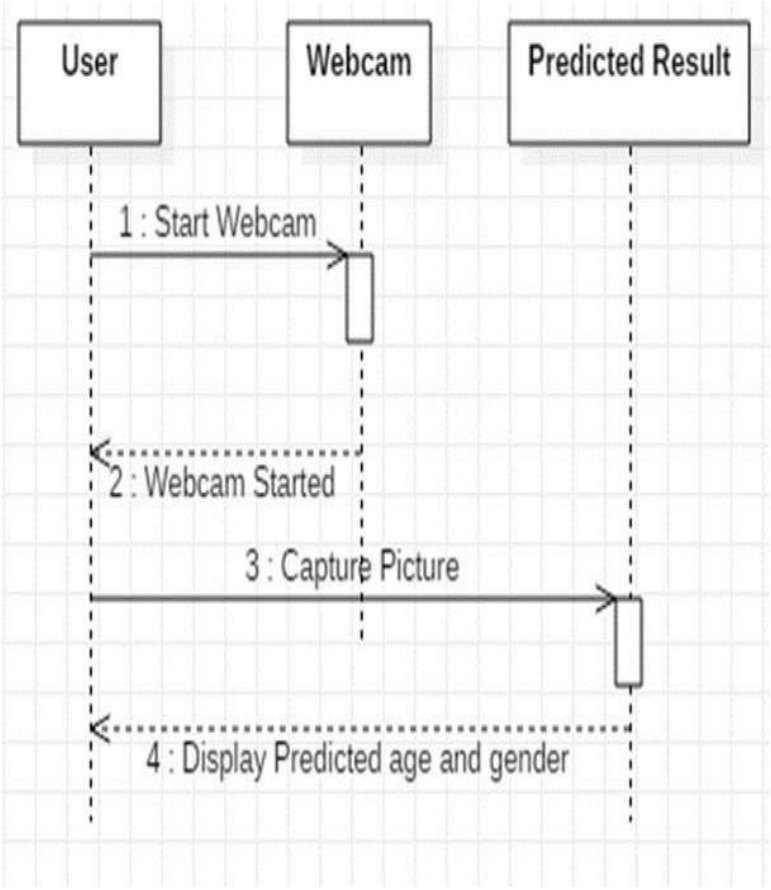
A use case diagram at its simplest is a representation of a user's interaction with thesystem anddepictingthespecificationsofausecase.Ausecasediagramcanportraythedifferenttypesof users of a system and the various ways that they interact with the system. This type of diagram is typicallyusedinconjunctionwiththetextualusecaseandwilloftenbeaccompaniedbyothertypes of diagrams aswell.

Usecasediagramisoneofthemanditsspecificpurposeistogather systemrequirementsandactors. Use case diagrams specify the events of a system and their flows. But use case diagramnever describes how they areimplemented.



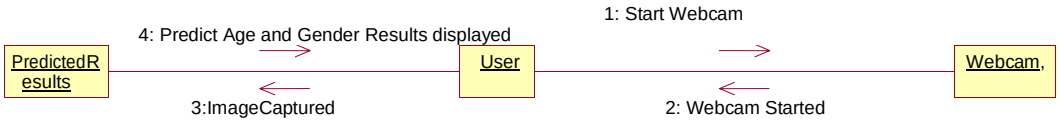
Sequence diagram

Asequencediagramisakindofinteractiondiagramthatshowshowprocessesoperatewith one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows objectinteractions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams, event scenarios, andtimingdiagrams.



Collaboration diagram

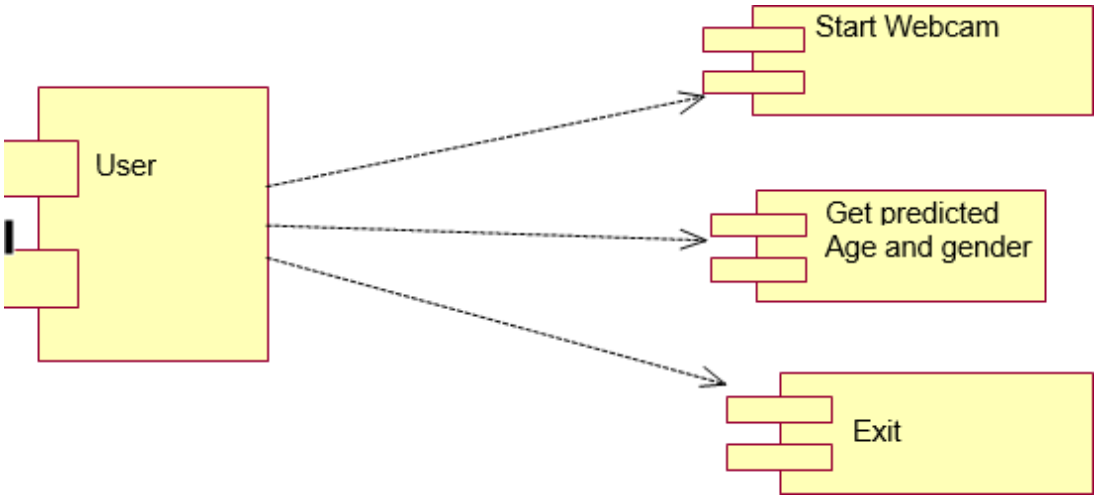
A collaboration diagram describes interactions among objects in terms of sequenced messages. Collaboration diagrams represent a combination of information taken from class, sequence, and use case diagrams describing both the static structure and dynamic behavior of a system.



Component Diagram

In the Unified Modeling Language, a component diagram depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems.

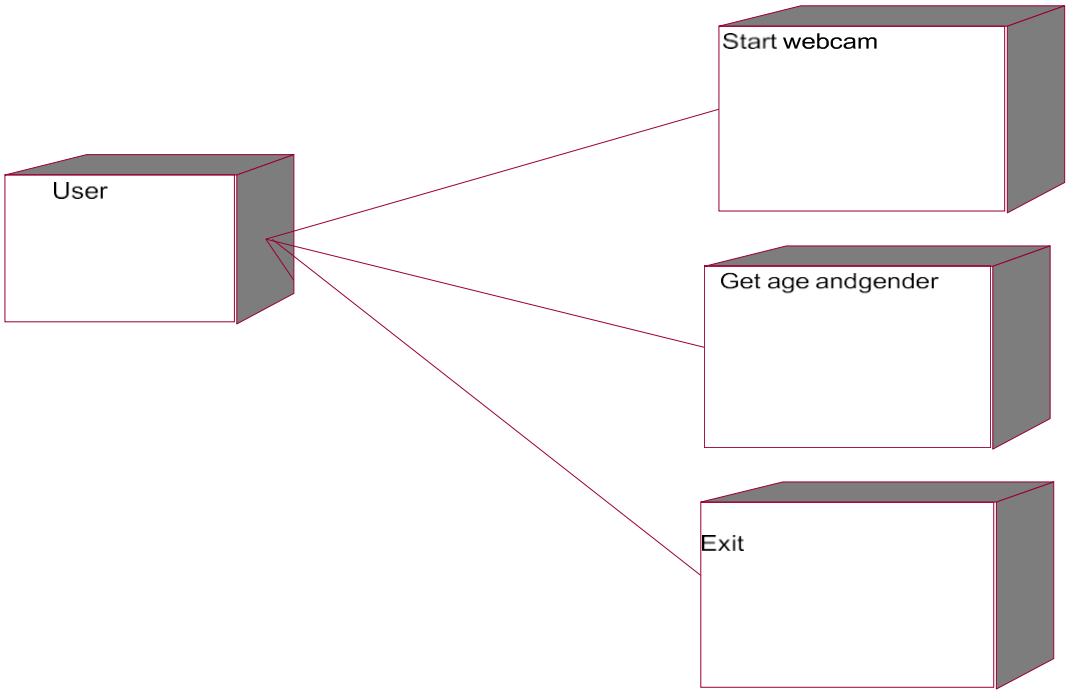
Components are wired together by using an assembly connector to connect the required interface of one component with the provided interface of another component. This illustrates the service consumer service provider relationship between the two components.


Start Webcam
User
Get predicted
Age and gender
Exit

Deployment diagram

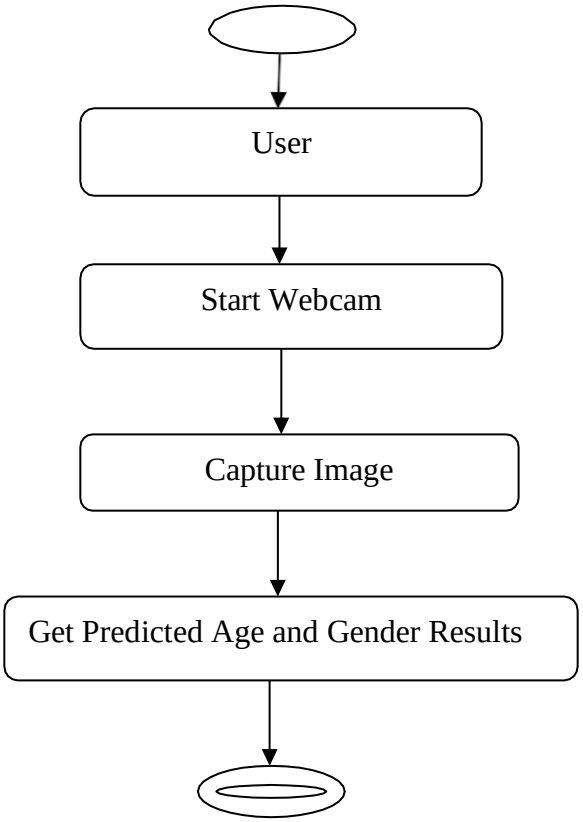
A deployment diagram in the Unified Modeling Language models the physical deploymentofartifactsonnodes.Todescribeawebsite,forexample,adeploymentdiagramwould show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (eg. JDBC, REST,RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within theboxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagrammay conceptually represent multiple physical nodes, such as a cluster of database servers.



Activity Diagram

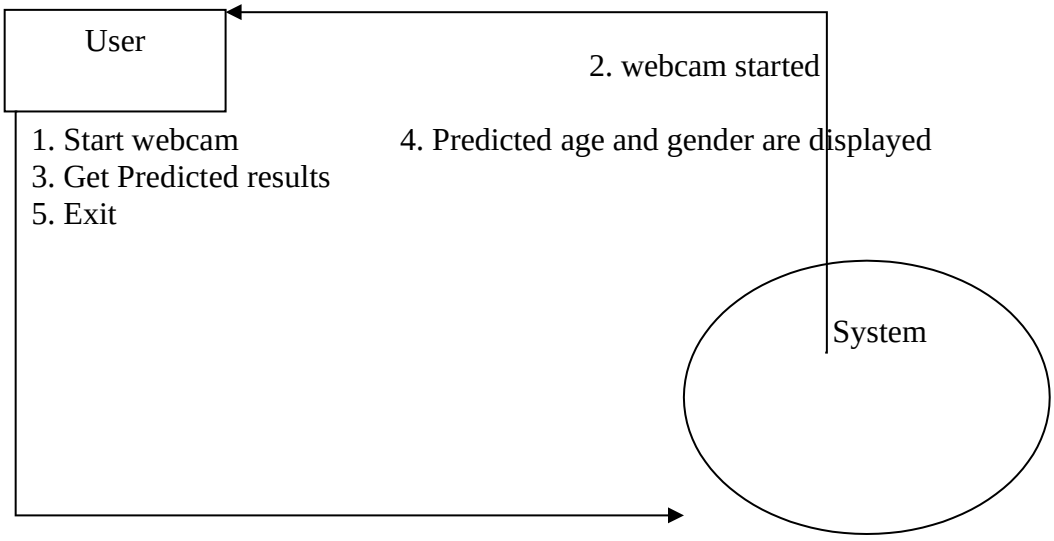
Activity diagram is another important diagram in UML to describe dynamic aspects of the system.It is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. Thisflow can be sequential, branched or concurrent



4.4 Dataflowdiagram

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs.Dataflowdiagramscanbeusedtoprovideaclearrepresentationofanybusinessfunction. The techniquestarts with an overall picture of the business and continues by analyzing each of the functionalareasofinterest.Thisanalysiscanbecarriedoutinpreciselythelevelofdetailrequired. The technique exploits amethod called top-down expansion to conduct the analysis in a targeted way.

As the name suggests, Data Flow Diagram (DFD) is an illustration that explicates the passage of information in a process. A DFD can be easily drawn using simple symbols. Additionally, complicated processes can be easily automated by creating DFDs using easy-to-use, free downloadable diagramming tools. A DFD is a model for constructing and analyzing information processes. DFD illustrates the flow of information in a process depending upon the inputs and outputs. A DFD can also be referred to as a Process Model. A DFD demonstrates business or technicalprocesswiththesupportoftheoutsidedatasaved,plusthedataflowingfromtheprocess to another and the endresults.



Chapter 5

Implementation

5.1 Python

Definition of python:

Python is a general-purpose language. It has wide range of applications from Web development (like: Django and Bottle), scientific and mathematical computing (Orange, SymPy, NumPy) to desktop graphical user Interfaces (Pygame, Panda3D). The syntax of the language is clean and length of the codeis relatively short. It's fun to work in Python because it allows you to think about the problem rather thanfocusing on the syntax.

History of Python:

Python is a fairly old language created by Guido Van Rossum. The design began in the late 1980sandwasfirstreleasedinFebruary1991asasuccessortoABCprogramminglanguage,which was inspired bySETL, capable of exception handling and interfacing with the Amoeba operating system. Its implementation began in December 1989.

Why Python was created?

In late 1980s, Guido Van Rossum was working on the Amoeba distributed operating system group. He wanted to use an interpreted language like ABC (ABC has simple easy-to-understand syntax) that could access the Amoeba system calls. So, he decided to create alanguage that was extensible. This led to design of a new language which was later namedPython.

Why the name Python?

> It wasn't named after a dangerous snake. Rossum was fan of a comedy series from late seventies.

The name "Python" was adopted from the same series "Monty Python's Flying Circus".

Features of python:

- **A simple language which is easier tolearn**

Pythonhasaverysimpleandelegantsyntax.It'smucheasiertoreadandwritePython programs compared to other languages like: C++, Java, C#. Python makes programming fun and allowsyoutofocusonthesolutionratherthansyntaxIfyouareanewbie,it'sagreatchoicetostart your journeywithPython.

- **Free andopen-source**

You can freely use and distribute Python, even for commercial use. Not only can you use and distribute software written in it, you can even make changes to the Python's source code. Python has a largecommunity constantly improving it in each iteration.

- **Portability**

YoucanmovePythonprogramsfromoneplatformtoanother,andrunitwithoutany changes. It runs seamlessly on almost all platforms including Windows, Mac OS X andLinux.

- **Extensible andEmbeddable**

Suppose an application requires high performance. You can easily combine pieces of C/C++or other languages with Python code. This will give your application high performance as well as scriptingcapabilities which other languages may not provide out of the box.

- **A high-level, interpretedlanguage**

Unlike C/C++, you don't have to worry about daunting tasks like memory management, garbage collection and so on. Likewise, when you run Python code, it automatically converts your code tothe language your computer understands. You don't need to worry about any lower-level operations.

5.2 Computervision

OpenCV is short for Open Source Computer Vision. Intuitively by the name, it is an open- source Computer Vision and Machine Learning library. This library is capable of processing real-timeimage and video while also boasting analytical capabilities. It supports the Deep Learning frameworks TensorFlow, Caffe, and PyTorch.

What is a CNN?

A Convolutional Neural Network is a deep neural network (DNN) widely used for the purposes of imagerecognition and processing and NLP. Also known as a ConvNet, a CNN has input and output layers, andmultiple hidden layers, many of which are convolutional. In a way, CNNs are regularized multilayer perceptrons.

Tobuildagenderandagedetectorthatcanapproximatelyguessthegenderandage of theperson (face) in a picture using Deep Learning on the Adiencedataset.

In this Python Project, we will use Deep Learning to accurately identify the gender and age of a person from a single image of a face. We will use the models trained by Tal Hassner and GilLevi. The predictedgender may be one of 'Male' and 'Female', and the predicted age may be one of the following ranges- (0– 2), (4 – 6), (8 – 12), (15 – 20), (25 – 32), (38 – 43), (48 – 53), (60 –

100) (8 nodes in the final softmax layer). It is very difficult to accurately guess an exact age from a single image because of factors likemakeup, lighting, obstructions, and facial expressions. And so, we make this a classification probleminstead of making it one ofregression.

The CNN Architecture

The convolutional neural network for this python project has 3 convolutional layers:

- Convolutional layer; 96 nodes, kernel size7
- Convolutional layer; 256 nodes, kernel size5
- Convolutional layer; 384 nodes, kernel size3

It has 2 fully connected layers, each with 512 nodes, and a final output layer of softmax type.

Softmax.

At the top of the proposed architecture lies a softmax layer, which computes the loss term that is optimized during training and also the class probabilities during a classification.

WhilesomelosslayerslikemulticlassSVMlosstreattheoutputofthefinalfullyconnected layeras the class scores, softmax (also known as multinomial logistic regression) treats thesescoresastheunnormalized logprobabilitiesoftheclasses.Thatis,ifwehaveziisthe score assigned to class i after the final fully connected layer, then the softmax functionis

$$fj(z) = ezj / k\ ezk$$

Becausewewanttomaximizetheloglikelihoodofthecorrectclass,thetermwewant to minimize is the negative loglikelihood.

$$Li = -log(efyi / j\ efj)$$

Because the softmax function takes real-valued scores being output from f and normalizes themby their exponentiated sum, it guarantees that the sum of all softmax scores is 1, thereby allowing it to be interpreted as a true class probability. It should be noted that the softmax loss isactually a particular form of a cross-entropy loss. More specifically, the cross-entropy between an actual distribution p and an approximate distribution q is defined as

$$H(p,q) = -X\ x\ p(x\ )logq(x)$$

From this it can be seen that the softmax classifier is really just minimizing the cross-entropybetween the estimated class probabilities and the real distribution, which would look like 1 predicted for the actual class and 0 predicted for everything else.

Stochastic Gradient Descent.

Now that we know how to calculate the loss, we need to know how to minimize it in order totrain an accurate classifier. The type of optimization used in this experiment is stochastic gradient descent. In order to explain this, first I will elaborate on the more generalizedformofgradientdescent.Thegradientofafunctionisreallyjustitsderivative, and therefore by definition it is the direction of steepest ascent (or descent if you move backwards alongit).

Thereforeifwecomputethegradientofthelossfunctionwithrespecttoallofthesystem variables/weights (in CNNs there can be up to millions of these), we will have the direction along which we can move toward our minimum loss most quickly byfollowing the negative ofthe gradient. Each time we compute the gradient we take a small step (governedbya hyperparameter)intheoppositedirection,andwere-evaluatetheloss,re-computethegradient,andrepeat.Thehope(andinfactthereality)isthatbyrepeatingthis process we will iterativelydecrease our loss function, which is reflective of the model becoming iteratively better at its classification task. Mathematically, we can write thisas

$$w = w \_ rwL$$

where n is the learning rate, also sometimes called the step size and rwL is the gradient of theloss term with respect to the weight vector w.

While this is theoretically great, the truth is that computing the gradient across the entire training set in order to make an incremental update to the weights is prohibitively computationally expensive. Therefore, alternate approaches have been invented that evaluate the gradient of the loss function over a sample of the training data, and use that approximate gradientto make the update. The reason this gradient is approximate is that although it is the optimal direction to travel down given the sample of images it was computed over, there is no telling what kinds of images it did not look at when computing thegradient.Therefore,makingthisformofmini-batchgradientdescent,asitiscalled,will still usually reach the minimum loss over time(or at least a local minimum), but it will require many more iterations onaverage.

5.3 Steps of age and genderdetection

The contents of our project are:

- opencv_face_detector.pbtxt
- opencv_face_detector.pb
- age_deploy.prototxt
- age_net.caffemodel
- gender_deploy.prototxt
- gender_net.caffemodel
- a few pictures to try the projecton

For face detection, we have a .pb file- this is a protobuf file (protocol buffer); it holds thegraph definition and the trained weights of the model. We can use this to run the trained model.Andwhilea.pbfileholdstheprotobufinbinaryformat,onewiththe.pbtxtextension holds it in textformat. These are TensorFlow files. For age and gender, the prototxt files describe the network configuration and the .caffemodel file defines theinternal states of the parameters of thelayers.

1. We use the argparse library to create an argument parser so we can get the image argument from the command prompt. We make it parse the argument holdingthe path tothe image to classify gender and agefor.
2. For face, age, and gender, initialize protocol buffer andmodel.
3. Initialize the mean values for the model and the lists of age ranges and genders toclassifyfrom.
4. Now, use the readNet() method to load the networks. The first parameterholds trainedweights and the second carries networkconfiguration.

Let’s capture video stream in case you’d like to classify on a webcam’s stream. Setpadding to 20

5. Now until any key is pressed, we read the stream and store the content into the nameshasFrame and frame. If it isn't a video, it must wait, and so we call up waitKey()fromcv2, thenbreak.
6. Let's make a call to the highlightFace() function with the faceNet and frame parameters, and what this returns, we will store in the names resultImg and faceBoxes. And if we got0 faceBoxes, it means there was no face to detect. Here, net is faceNet- this model is the DNN Face Detector and holds only about 2.7MBondisk.
    - Create a shallow copy of frame and get its height andwidth.
    - Create a blob from the shallowcopy.
    - Set the input and make a forward pass to thenetwork.
    - faceBoxes is an empty list now. for each value in 0 to 127, define the confidence (between 0 and 1). Wherever we find the confidence greater thantheconfidencethreshold, whichis0.7, wegetthex1,y1, x2, andy2 coordinates and append a listof those tofaceBoxes.
    - Then, we put up rectangles on the image for each such list of coordinates andreturn two things: the shallow copy and the listof faceBoxes.
7. But if there are indeed faceBoxes, for each of those, we define theface, create a 4- dimensional blob from the image. In doing this, we scale it, resize it, and pass in themeanvalues.
8. We feed the input and give the network a forward pass to get the confidenceof the twoclass. Whichever is higher, that is the gender of the person in the picture.
9. Then, we do the same thing forage.
10. We'll add the gender and age texts to the resulting image and display it withimshow().

5.4 Sourcecode

5.4.1 detect.py

```
import cv2
import math
import argparse

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

    net.setInput(blob)
    detections=net.forward()
    faceBoxes=[]
    for i inrange(detections.shape[2]):
        confidence=detections[0,0,i,2]
        ifconfidence>conf_threshold:
            x1=int(detections[0,0,i,3]*frameWidth)
            y1=int(detections[0,0,i,4]*frameHeight)
        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)Width)

            y1=int(detections[0,0,i,4]*frameHeight)
            x2=int(detections[0,0,i,5]*frameWidth)
            y2=int(detections[0,0,i,6]*frameHeight)
            faceBoxes.append([x1,y1,x2,y2])
            cv2.rectangle(frameOpencvDnn,(x1,y1),(x2,y2),(0,255,0),int(round(frameHeight/150)),8)
        returnframeOpencvDnn,faceBoxes
```

```
parser=argparse.ArgumentParser()
parser.add_argument('--image')
args=parser.parse_args()
faceProto="C:/Users/Hi/Documents/Datagain/Age and Gender/Age/opencv_face_detector.pbtxt"
faceModel="C:/Users/Hi/Documents/Datagain/Age and Gender/Age/opencv_face_detector_uint8.pb"
ageProto="C:/Users/Hi/Documents/Datagain/Age and Gender/Age/age_deploy.prototxt"
ageModel="C:/Users/Hi/Documents/Datagain/Age and Gender/Age/age_net.caffemodel"

genderProto="C:/Users/Hi/Documents/Datagain/Age and Gender/Age/gender_deploy.prototxt"
genderModel="C:/Users/Hi/Documents/Datagain/Age and Gender/Age/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']
faceNet=cv2.dnn.readNet(faceModel,faceProto)
ageNet=cv2.dnn.readNet(ageModel,ageProto)
genderNet=cv2.dnn.readNet(genderModel,genderProto)
video=cv2.VideoCapture(args.image if args.image else 0)
padding=20

while cv2.waitKey(1)<0:
    hasFrame,frame=video.read()if
    not hasFrame:
        cv2.waitKey()
        break
```

```
    resultImg,faceBoxes=highlightFace(faceNet,
    frame)if not faceBoxes:
print("No face
detected")for faceBox
in faceBoxes:
        face=frame[max(0,faceBox[1]-padding):
                min(faceBox[3]+padding,frame.shape[0]-1),max(0,faceBox[0]-
                padding)
                :min(faceBox[2]+padding, frame.shape[1]-1)]
      blob=cv2.dnn.blobFromImage(face, 1.0, (227,227), MODEL_MEAN_VALUES,
      swapRB=False)genderNet.setInput(blob)
        genderPreds=genderNet.forward()
        gender=genderList[genderPreds[0].ar
        gmax()]print(f'Gender: {gender}')
        ageNet.setInput(blob)
        agePreds=ageNet.forward()
        age=ageList[agePreds[0].argmax()]
        print(f'Age: {age[1:-1]} years')
        cv2.putText(resultImg,f'{gender},
        {age}', (faceBox[0],faceBox[1]-10),
        cv2.FONT_HERSHEY_SIMPLEX,
        0.8, (0,255,255), 2,cv2.LINE_AA
         cv2.imshow("Detecting age and gender", resultImg)
```

5.4.2 gender_deploy.prototxt

```
name: "CaffeNet"
input: "data"
input_dim: 10
input_dim: 3
input_dim: 227
input_dim: 227
layers {
  name: "conv1"
  type: CONVOLUTION
  bottom: "data"
  top: "conv1"
  convolution_param {
    num_output: 96
    kernel_size: 7
    stride: 4
  }
}
layers {
  name: "relu1"
  type: RELU
  bottom: "conv1"
  top: "conv1"
}
layers {
  name: "pool1"
  type: POOLING
  bottom: "conv1"
  top: "pool1"
  pooling_param {
  pool: MAX
  kernel_size: 3
    stride: 2
}
}
```

```
layers {
  name: "norm1"
  type: LRN
  bottom: "pool1"
  top: "norm1"
  lrn_param {
  local_size: 5
    alpha: 0.0001
    beta: 0.75
  }
}
layers {
  name: "conv2"
  type: CONVOLUTION
  bottom: "norm1"
  top: "conv2"
  convolution_param {
  num_output: 256
    pad: 2
    kernel_size: 5
  }
}
layers {
  name: "relu2"
  type: RELU
  bottom: "conv2"
  top: "conv2"
}
layers {
  name: "pool2"
  type: POOLING
  bottom: "conv2"
  top: "pool2"
  pooling_param {
  pool: MAX
  kernel_size: 3
    stride: 2
  }
```

```
layers {
  name: "drop7"
  type: DROPOUT
  bottom: "fc7"
  top: "fc7"
  dropout_param {
    dropout_ratio: 0.5
  }
}
layers {
  name: "fc8"
  type: INNER_PRODUCT
  bottom: "fc7"
  top: "fc8"
  inner_product_param {
    num_output: 2
  }
}
layers {
  name: "prob"
  type: SOFTMAX
  bottom: "fc8"
  top: "prob"
}
```

5.4.3 age_deploy.prototext

```
name: "CaffeNet"
input: "data"
input_dim: 1
input_dim: 3
input_dim: 227
input_dim: 227
layers {
  name: "conv1"
  type: CONVOLUTION
  bottom: "data"
  top: "conv1"
  convolution_param {
    num_output: 96
    kernel_size: 7
    stride: 4
  }
}
layers {
  name: "relu1"
  type: RELU
  bottom: "conv1"
  top: "conv1"
}
layers {
  name: "conv2"
  type: CONVOLUTION
  bottom: "norm1"
  top: "conv2"
  convolution_param {
  num_output: 256
    pad: 2
    kernel_size: 5
  }
}
```

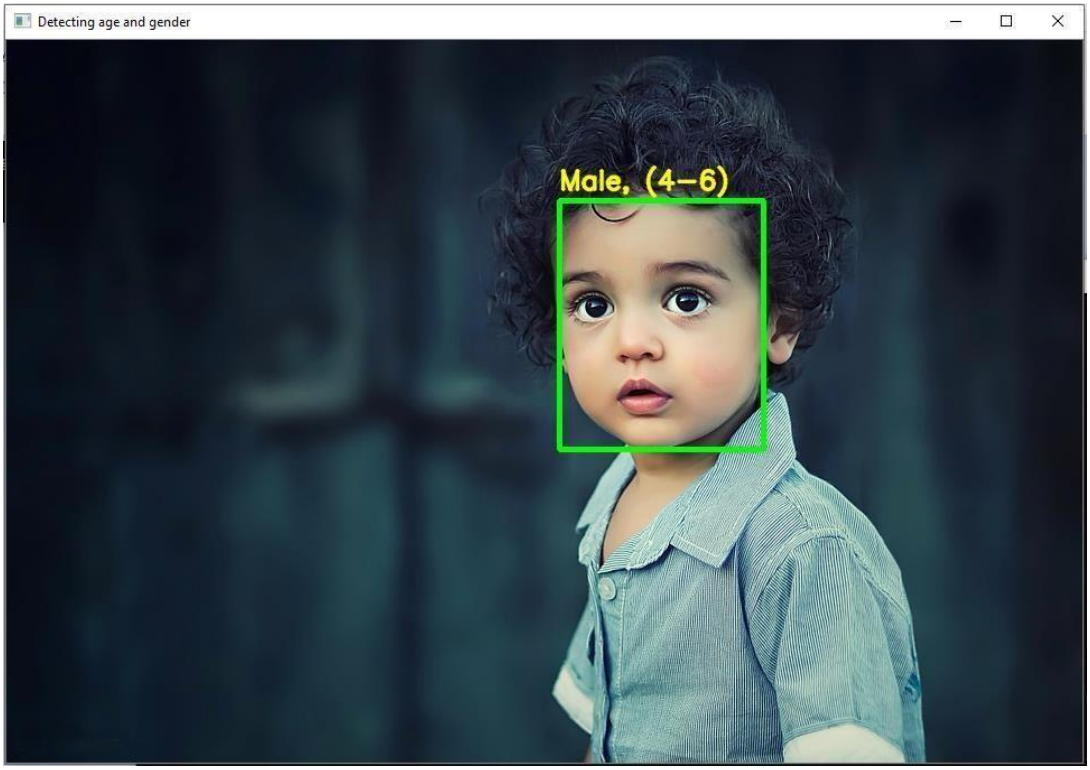
```
layers {
  name: "relu2"
  type: RELU
  bottom: "conv2"
  top: "conv2"
}
layers {
  name: "pool2"
  type: POOLING
  bottom: "conv2"
  top: "pool2"
  pooling_param {
  pool: MAX
  kernel_size: 3
    stride: 2
  }
}
layers {
  name: "norm2"
  type: LRN
  bottom: "pool2"
  top: "norm2"
  lrn_param {
  local_size: 5
    alpha: 0.0001
    beta: 0.75
  }
}
layers {
  name: "conv3"
  type: CONVOLUTION
  bottom: "norm2"
  top: "conv3"
  convolution_param {
  num_output: 384
    pad: 1
    kernel_size: 3
  }
}
```

```
layers{
  name: "relu3"
  type: RELU
  bottom: "conv3"
  top: "conv3"
}
layers {
  name: "pool5"
  type: POOLING
  bottom: "conv3"
  top: "pool5"
  pooling_param {
  pool: MAX
  kernel_size: 3
    stride: 2
  }
}
layers {
  name: "fc6"
  type: INNER_PRODUCT
  bottom: "pool5"
  top: "fc6"
  inner_product_param {
    num_output: 512
  }
}
layers {
  name: "relu6"
  type: RELU
  bottom: "fc6"
  top: "fc6"
}
layers {
  name: "drop6"
  type: DROPOUT
  bottom: "fc6"
  top: "fc6"
  dropout_param {
    dropout_ratio: 0.5
  }
}
```

```
layers {
  name: "relu7"
  type: RELU
  bottom: "fc7"
  top: "fc7"
}
layers {
  name: "drop7"
  type: DROPOUT
  bottom: "fc7"
  top: "fc7"
  dropout_param {
    dropout_ratio: 0.5
  }
}
layers {
  name: "fc8"
  type: INNER_PRODUCT
  bottom: "fc7"
  top: "fc8"
  inner_product_param {
    num_output: 8
  }
}
layers {
  name: "prob"
  type: SOFTMAX
  bottom: "fc8"
  top: "prob"

}
```

5.5 Examples ofoutput



Chapter 6

Testing

6.1 Implementation andtesting

Implementation and Testing:

Implementation is one of the most important tasks in project is the phase in which one has to be cautions because all the efforts undertaken during the project will be very interactive. Implementation is the most crucial stage in achieving successful system and giving the users confidencethatthenewsystemisworkableandeffective.Eachprogramistestedindividuallyatthe time of development using the sampledata and has verified that these programs link together inthe way specified in the program specification. The computer system and its environment are testedto the satisfaction of theuser.

Implementation

The implementation phase is less creative than system design. It is primarily concernedwithusertraining,andfileconversion.Thesystemmayberequiringextensiveuser training. The initial parameters of the system should be modify as a result of a programming. Asimpleoperatingprocedureisprovidedsothattheusercanunderstandthedifferentfunctions clearly and quickly. Thedifferent reports can be obtained either on the inkjet or dot matrix printer, which is available at the disposal of the user. The proposed system is very easy to implement. In general implementation is used to mean the process of converting a new or revised system design into an operationalone.

Testing

Testingistheprocesswherethetestdataispreparedandisusedfortestingthemodules individually and later the validation given for the fields. Then the system testing takes placewhich makessure that all components of the system property function as a unit. Thetest

data should be chosen such that it passed through all possible condition. Actually, testing is the state of implementation which aimed at ensuring that the system works accurately and efficiently before following is the description of the testing strategies, which were carried out during the testing period.

System Testing

Testing has become an integral part of any system or project especially in the field of informationtechnology.Theimportanceoftestingisamethodofjustifying,ifoneisreadytomove further, be it to becheck if one is capable to with stand the rigors of a particular situation cannotbe underplayed and that is why testing before development is so critical. When the software is developed before it is given to user touser the software must be tested whether it is solving the purposeforwhichitisdeveloped.Thistestinginvolvesvarioustypesthroughwhichonecanensure the software is reliable. The program was tested logically and pattern of execution of the program for a set of data are repeated. Thus, the code was exhaustively checked for all possible correctdata and the outcomes were alsochecked.

Module Testing

To locate errors, each module is tested individually. This enables us to detect errorandcorrectitwithoutaffectinganyothermodules.Whenevertheprogramisnotsatisfyingthe required function, it must be corrected to get the required result. Thus, all the modules are individuallytestedfrombottomupstartingwiththesmallestandlowestmodulesandproceedingto the next level. Each module inthe system is tested separately. For example, the job classification module is tested separately. This moduleis tested with different job and its approximate execution time and the result of the test is compared with the results that are prepared manually. The comparisonshowsthattheresultsproposedsystemworksefficientlythantheexistingsystem.Each module in the system is tested separately. In this system the resource classification and job schedulingmodulesaretestedseparatelyandtheircorrespondingresultsareobtainedwhichreduces the process waitingtime.

Integration Testing

Afterthemoduletesting,theintegrationtestingisapplied.Whenlinkingthemodules theremaybechanceforerrorstooccur,theseerrorsarecorrectedbyusingthistesting.Inthissystem allmodulesareconnectedandtested.Thetestingresultsareverycorrect.Thus,themappingofjobs with resources is done correctly by thesystem.

Acceptance Testing

When that user fined no major problems with its accuracy, the system passers through a final acceptance test. This confirms that the system needs the original goals, objectives and requirements established during analysis without execution which elimination wastage of timeand moneyacceptancetestsontheshouldersofusersandmanagement,itisfinallyacceptableandready for theoperation.

Test Case Id	Test Case Name	Test Case Desc.	Test Steps			Test Case Status	Test Priority
			Step	Expected	Actual		
01	Start webcam	Test whether the webcam is started or not	If the webcam may not started	We cannot do the further operations	The webcam started successfully	High	High
02	Predict results	Verify the predicted results displayed or not	Without capturing the user image	We cannot get the age and gender predicted results	This displays age and gender successfully	High	High

6.2 White boxtesting

White Box Testing is a testing in which the software tester has knowledge of the innerworkings,structureandlanguageofthesoftware,oratleastitspurpose.Ithasapurpose. It is used totest areas that cannot be reached from a black boxlevel.

White Box Penetration Testing:

Inthistesting,thetester/developerhasfullinformationoftheapplication'ssourcecode, detailednetwork information, IP addresses involved and all server information the application runs on. The aim is to attack the code from several angles to expose securitythreats.

White Box Mutation Testing:

Mutation testing is often used to discover the best coding techniques to use for expanding a softwaresolution.

White Box Testing Tools

Below is a list of top white box testing tools.

- ParasoftJtest
- EclEmma
- NUnit
- PyUnit
- HTMLUnit
- CppUnit

Advantages of White Box Testing

- Code optimization by finding hiddenerrors.
- White box test cases can be easilyautomated.
- Testing is more thorough as all code paths are usuallycovered.
- Testing can start early in SDLC even if GUI is notavailable.

Disadvantages of White Box Testing

- White box testing can be quite complex andexpensive.
- Developers who usually execute white box test cases detest it. The white box testing bydevelopers is not detailed and can lead to productionerrors.

6.3 Black boxtesting

It is defined as a testing technique in which functionality of the Application Under Test (AUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on softwarerequirementsandspecifications.InBlackBoxTestingwejustfocusoninputsand output of the software system withoutbothering about internal knowledge of the software program.

TheaboveBlack-Boxcanbeanysoftwaresystemyouwanttotest.ForExample,an operatingsystem like Windows, a website like Google, a database like Oracle or even your own custom application Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal codeimplementation.

How to do black box testing?

Here are the generic steps followed to carry out any type of Black Box Testing.

- Initially, the requirements and specifications of the system areexamined.
- Tester chooses valid inputs (positive test scenario) to check whether SUTprocesses them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the SUT isable to detectthem.
- Tester determines expected outputs for all thoseinputs.
- Software tester constructs test cases with the selectedinputs.
- The test cases areexecuted.
- Software tester compares the actual outputs with the expectedoutputs.
- Defects if any are fixed and re-tested.

Types of Black Box Testing

There are many types of Black Box Testing but the following are the prominent ones -

- **Functional testing** - This black box testing type is related to the functional requirementsof a system; it is done by softwaretesters.
- **Non-functional testing** - This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance,scalability,usability.
- **Regression testing** - Regression Testing is done after code fixes, upgrades or anyothers system maintenance to check the new code has not affected the existingcode.

Blackbox testing techniques

Following are the prominent Test Strategy amongst the many used in Black box Testing

Equivalence Class Testing: It is used to minimize the number of possible test cases to an optimumlevel while maintains reasonable test coverage.

Boundary Value Testing: Boundary value testing is focused on the values at boundaries. This technique determines whether a certain range of values are acceptable by the system or not. It is very useful in reducing the number of test cases. It is most suitable for the systems where an input is withincertainranges.

Decision Table Testing: A decision table puts causes and their effects in a matrix. There is a unique Combination in each column.

Chapter 7

Conclusion and future scope

7.1 Conclusion

Though many previous methods have addressed the problems of age and genderclassification,untilrecently,muchofthisworkhasfocusedonconstrained images taken in lab settings. Such settings do not adequately reflect appearance variations common to the real-world images in social websites and online repositories.Internetimages,however,arenotsimplymorechallenging:theyare also abundant. The easy availability of huge image collections provides modern machinelearningbasedsystemswitheffectivelyendlesstrainingdata,thoughthis data isnot always suitably labeled for supervised learning. Taking example from the related problem of face recognition we explore how well deep CNN perform on these tasks using Internet data. We provide results with a lean deep-learning architecture designed to avoid overfitting due to the limitation of limited labeled data. Our network is "shallow" compared to some of the recent network architectures, thereby reducing the number of its parameters and the chance for overfitting. We further inflate the size of the training data by artificially adding croppedversionsoftheimagesinourtrainingset.Theresultingsystemwastested on the Adience benchmark of unfiltered images and shown to significantly outperform recent state of the art. Two important conclusions can be made from our results. First, CNN can be used to provide improved age and gender classification results, even considering the much smaller size of contemporary unconstrained image sets labeled for age and gender. Second, the simplicity of ourmodelimpliesthat moreelaboratesystemsusing moretrainingdatamaywell be capable of substantially improving results beyond those reported here. We have proposed an sAM, which improves on the traditional AAM. When used for facialfeatureextraction,themodeldescribesthefacewithveryfewcomponents. For instance, we used just 13 components to effectively represent the face on FGNET-ADasopposedtoAAM,whichrequiresbetween50and200parameters. When usedfor age estimation, we achieved 5.49 MAE, which is comparable to most state-of-the-art algorithms and better than most algorithms that used AAM for featureextraction.

Inthiswork,itisbeenconcludedthatdetectionofageandgendertakeconsiderationofresearch few years ago. In this work, technique of morphological and SIFT is applied to search key featuresfromtheimages.Thekeyfeaturesoftheimagesarethecolorandtextureoftheimage. Thesimulationresultsshowsthatproposedalgorithmperformedwellintermsoffaultdetection rate and accuracy. In future, further improvement will be done in proposed work for iris reorganization for betterreorganization.

Overall study of contribution made on gender classification and age estimation can be used in to solve the real-time application problems. In this paper most of the research work done is in Convolutional Neural Networks. Eleven unlike types of neural networks have been discussed with their MAE and accuracy obtained by models. Additionally, function extraction in addition to distinction of a few functions is actually carried out using just an individual element extractor or maybe a one-time classifier along with in various additional works, fusion is actually followed to do distinction process or maybe attribute extraction. On the future direction, results that are Good for gender recognition as well as years opinion can continue to be received utilizing transfer learning strategies with expansion in reliability. Combos of fusions as well as datasets of attributes might be what is on the horizon for the development of rich learning and from 2D to 3D Facial Data. Moreover, ethnicity estimation, Affective behaviour analysis and numerous additional demographic features could be verified for the performance of them by the classifier of NeuralNetworks.

7.2 Futurescope

There are many possibilities in age and gender estimation research. A n immediate idea would be to look more deeply into training models with integral images as additional color channels using more varied neural network architectures. Another idea would be to use more varied neural network architecture specifically for gender prediction. Many current tools use the same architecture for both age and gender prediction.

In spite of the fact that numerous past techniques are developed regarding Gender and age prediction, they do not yield accurate outcomes. This is due to certain specifications like light conditions, alignment of face near webcam etc. Also, those previous methods can only detect few faces. Our proposed System rectifies all those above problems and produces accurate results. Our system can identify nearly around seven faces at a time. This improvement in performance is due to Wide Resnet architecture. The features like dropout, weight decay reduced the training time required for the network. Lastly, the simplicity of our model infers that more intricate systems using more training data may well be capable of substantially improving results beyond those reported here.

Chapter8

References

[1]T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. Trans. Pattern Anal. Mach. Intell., 28(12):2037–2041, 2006. 2

[2] S. Baluja and H. A. Rowley. Boosting sex identification performance. Int. J.Comput. Vision, 71(1):111–119, 2007. 2

[3] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In Int. Conf. Mach. Learning, volume 3, pages 11–18, 2003.2

[4] W.-L. Chao, J.-Z. Liu, and J.-J. Ding. Facial age estimation based on label-sensitive learning and age-oriented regression. Pattern Recognition, 46(3):628–641, 2013. 1,2

[5] K.Chatfield,K.Simonyan,A.Vedaldi,andA.Zisserman.Returnofthedevilinthedetails: Delving deep into convolutional nets. arXiv preprint arXiv:1405.3531, 2014. 3

[6] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao. Wld: A robust local image descriptor. Trans. Pattern Anal. Mach. Intell., 32(9):1705–1720, 2010.2

[7] S.E.Choi,Y.J.Lee,S.J.Lee,K.R.Park,andJ.Kim.Ageestimationusingahierarchical classifier based on global and local facial features. Pattern Recognition, 44(6):1262–1281, 2011.2

[8] T.F.Cootes,G.J.Edwards,andC.J.Taylor.Activeappearancemodels.InEuropeanConf. Comput. Vision, pages 484–498. Springer, 1998. 2

[9] C. Cortes and V. Vapnik. Support-vector networks. Machine learning, 20(3):273–297, 1995.2

[10] E.Eidinger,R.Enbar,andT.Hassner.Ageandgenderestimationofunfilteredfaces. Trans. on Inform. Forensics and Security, 9(12), 2014. 1, 2, 5, 6

[11] Y.Fu,G.Guo,andT.S.Huang.Agesynthesisandestimationviafaces:Asurvey.Trans. Pattern Anal. Mach. Intell., 32(11):1955–1976, 2010.2

[12] Y. Fu and T. S. Huang. Human age estimation with regression on discriminative aging manifold. Int. Conf. Multimedia, 10(4):578–584, 2008.2

[13] K. Fukunaga. Introduction to statistical pattern recognition. Academic press, 1991.2

[14] A. C. Gallagher and T. Chen. Understanding images of groups of people. In Proc.Conf. Comput. Vision Pattern Recognition, pages 256–263. IEEE, 2009. 2, 5

[15] F. Gao and H. Ai. Face age classification on consumer images with gabor feature and fuzzy lda method. In Advances in biometrics, pages 132–141. Springer, 2009. 1,2

[16] X.Geng,Z.-H.Zhou,andK.Smith-Miles.Automaticageestimationbasedonfacialaging patterns. Trans. Pattern Anal. Mach. Intell., 29(12):2234–2240, 2007.2

[17] B. A. Golomb, D. T. Lawrence, and T. J. Sejnowski. Sexnet: A neural network identifies sex from human faces. In Neural Inform. Process. Syst., pages 572–579, 1990. 2

[18] A.Graves,A.-R.Mohamed,andG.Hinton.Speechrecognitionwithdeeprecurrentneural networks. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pages 6645–6649. IEEE, 2013.3

[19] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang. Imagebased human age estimation by manifold learning and locally adjusted robust regression. Trans. Image Processing, 17(7):1178–1188, 2008.2

[20] G.Guo,G.Mu,Y.Fu,C.Dyer,andT.Huang.Astudyonautomaticageestimationusing a large database. In Proc. Int. Conf. Comput. Vision, pages 1986–1991. IEEE, 2009. 2 [21] H. Han, C. Otto, and A. K. Jain. Age estimation from face images: Human vs. machine performance. In Biometrics (ICB), 2013 International Conference on. IEEE, 2013.2

[21] T. Hassner. Viewing real-world faces in 3d. In Proc. Int. Conf. Comput. Vision, pages 3607–3614. IEEE, 2013. 6 [23] T. Hassner, S. Harel, E. Paz, and R. Enbar. Effective face frontalizationinunconstrainedimages.Proc.Conf.Comput.VisionPatternRecognition,2015. 5,6

[22] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. arXivpreprint

arXiv:1207.0580, 2012. 5

[23] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technicalreport, Technical Report 07-49, University of Massachusetts, Amherst, 2007. 3, 5

[24] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama,and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093, 2014. 5

[25] A.Karpathy,G.Toderici,S.Shetty,T.Leung,R.Sukthankar,andL.Fei-Fei.Large-scale videoclassificationwithconvolutionalneuralnetworks.InProc.Conf.Comput.VisionPattern Recognition, pages 1725–1732. IEEE, 2014.3

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Neural Inform. Process. Syst., pages 1097–1105, 2012. 3,4

[27] Y.H.KwonandN.daVitoriaLobo.Ageclassificationfromfacialimages.InProc.Conf. Comput. Vision Pattern Recognition, pages 762–767. IEEE, 1994. 1, 2

[28]A. Lanitis. The FG-NET aging database, 2002. Available: www-prima.inrialpes.fr/FGnet/html/ benchmarks.html.2

[29] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. Neural computation, 1(4):541–551, 1989. 1,3

[30] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher lineardiscriminantmodelforfacerecognition.Trans.ImageProcessing,11(4):467–476,2002. 1,2

AGE AND GENDER DETECTION

L N.Bhavani Priya[1],U. Sree Lakshmi[2], M.Prasanna[3], P.Dr.Santosh Sravanthi[4], IV B. Tech,
B.NagaEswari[5]Asst.professor Department of Information Technology
Vignan's Nirula Institute of Technology & Science for Women, Peda Palakaluru,
Guntur-522009, Andhra Pradesh, India.
nagaeswaribodapati@gmail.com

Abstract

Automatic age and gender classification has become relevant to an increasing number of applications, particularly since the rise of social platforms and social media. Nevertheless, performance of existing methods on real-world images is still significantly lacking, especially when compared to the tremendous leaps in performance recently reported for the related task of face recognition. In this paper we show that by learning representations through the use of deep-convolutional neural networks (CNN), a significant increase in performance can be obtained on these tasks. To this end, we propose a simple convolutional net architecture that can be used even when the amount of learning data is limited. We evaluate our method on the recent Adience benchmark for age and gender estimation and show it to dramatically outperform current state-of-the-artmethods.

1. Introduction

Age and gender play fundamental roles in social interactions. Language's reserve different salutations and grammar rules for men or women, and very often different vocabularies are used when addressing elders compared to young people. In this paper we show that by learning representations through the use of deep-convolutional neural networks (CNN), a significant increase in performance can be obtained on these tasks. To this end, we propose a simple convolutional net architecture that can be used even when the amount of learning data is limited. We evaluate our method on the recent Adience benchmark for age and gender estimation and show it to dramatically outperform current state-of-the-art methods. Like other branches of facial analysis, automatic aging and gender classification are hindered by a host of factors including illumination variation, facial expressions, and pose variation to mention but a few. Several approaches have been documented in the literature to circumvent these problems. Research on facial aging can be categorized into age estimation, age progression, and age invariant face recognition (AIFR). Age estimation refers to the automatic labelling of age groups or the specific ages of individuals using information obtained from theirfaces.

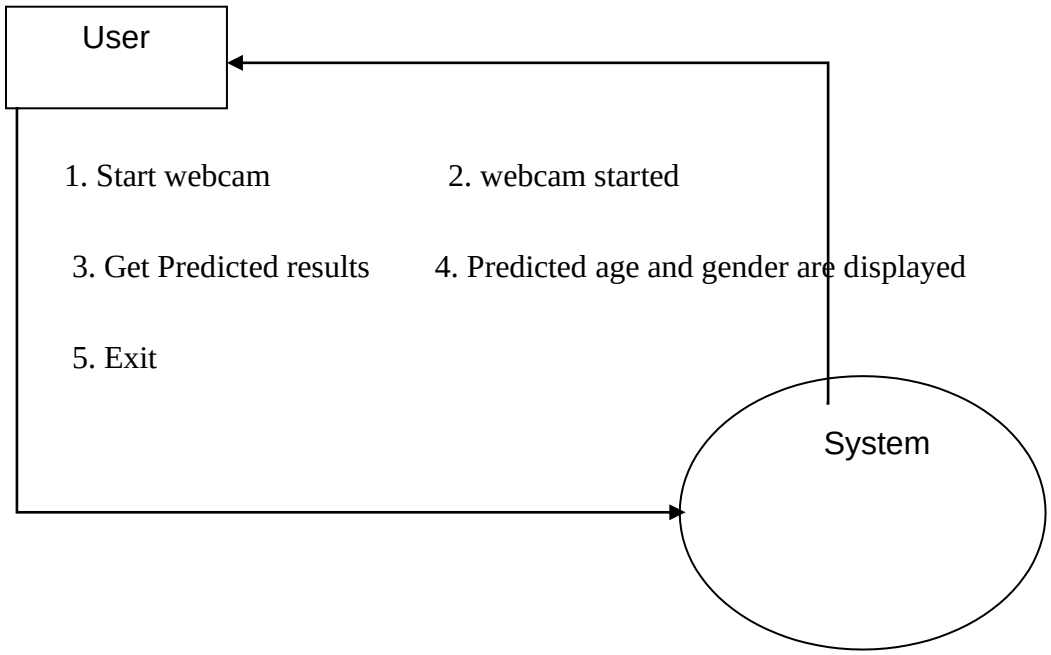
Keywords: CNN, Face recognition, regression, age and gender estimation

2. Proposedmodel

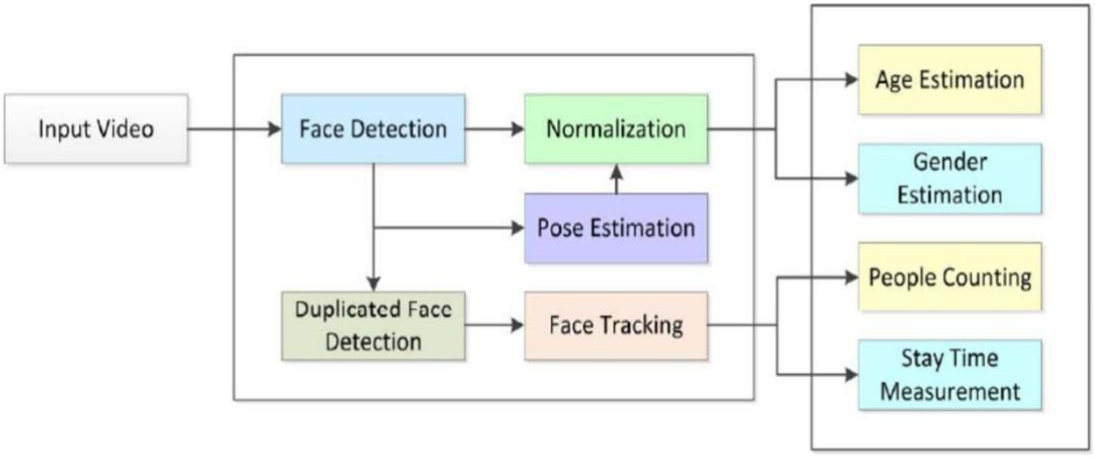
In this paper, we proposed an age and gender predicted framework by using deep-convolutional neural networks (CNN). Two important conclusions can be made from our results. First, CNN can be used to provide improved age and gender classification results, even considering the much smaller size of contemporary unconstrained image sets labelled for age and gender. Second, the simplicity of our model implies that more elaborate systems using more training data may well be capable of substantially improving results beyond those reported here.

3. Dataflowdiagram

As the name suggests, Data Flow Diagram (DFD) is an illustration that explicates the passage of information in a process. A DFD can be easily drawn using simple symbols. Additionally, complicated processes can be easily automated by creating DFDs using easy-to-use, free downloadable diagramming tools. A DFD is a model for constructing and analyzing information processes. DFD illustrates the flow of information in a process depending upon the inputs and outputs. A DFD can also be referred to as a Process Model. A DFD demonstrates business or technical process with the support of the outside data saved, plus the data flowing from the process to another and the end results.



4. Systemarchitecture



Face recognition and pose estimation:

In order to reduce processing time, face detection is performed for only every third input frame, and 64×64 is the minimum size of the face region used for detection. Facial images using localized eye positions are aligned to apply the face estimation algorithm. In addition, an automatic and monocular head pose estimation method is used for acquiring frontal faces using pitch, yaw and roll values. POSIT (Pose from Orthography and Scaling with ITerations), a hybrid poses estimation algorithm based on both algebraic and optimizing algorithms, is applied for head pose estimation. POSIT has advantages from both approaches and has good speed and accuracy. Thus, POSIT (Pose from Orthography and Scaling with ITerations), a hybrid poses estimation algorithm based on both algebraic and optimizing algorithms, is applied for head pose estimation.

Face tracking and people counting:

The people counting method based on face tracking consists of face detection, face tracking, tracking failure detection, data association and counting modules. The tracking framework is based on a multi-person tracking algorithm (Choi et al., 2015). An HSV (Hue Saturation Value) histogram is employed as an observation model, and Gaussian perturbation is employed as a motion model in a particle filter-based tracking framework. The people counting method based on face tracking consists of face detection, face tracking, tracking failure detection, data association and counting modules. The tracking framework is based on a multi-person tracking algorithm (Choi et al., 2015). An HSV (Hue Saturation Value) histogram is employed as an observation model, and Gaussian perturbation is employed as a motion model in a particle filter-based tracking framework.

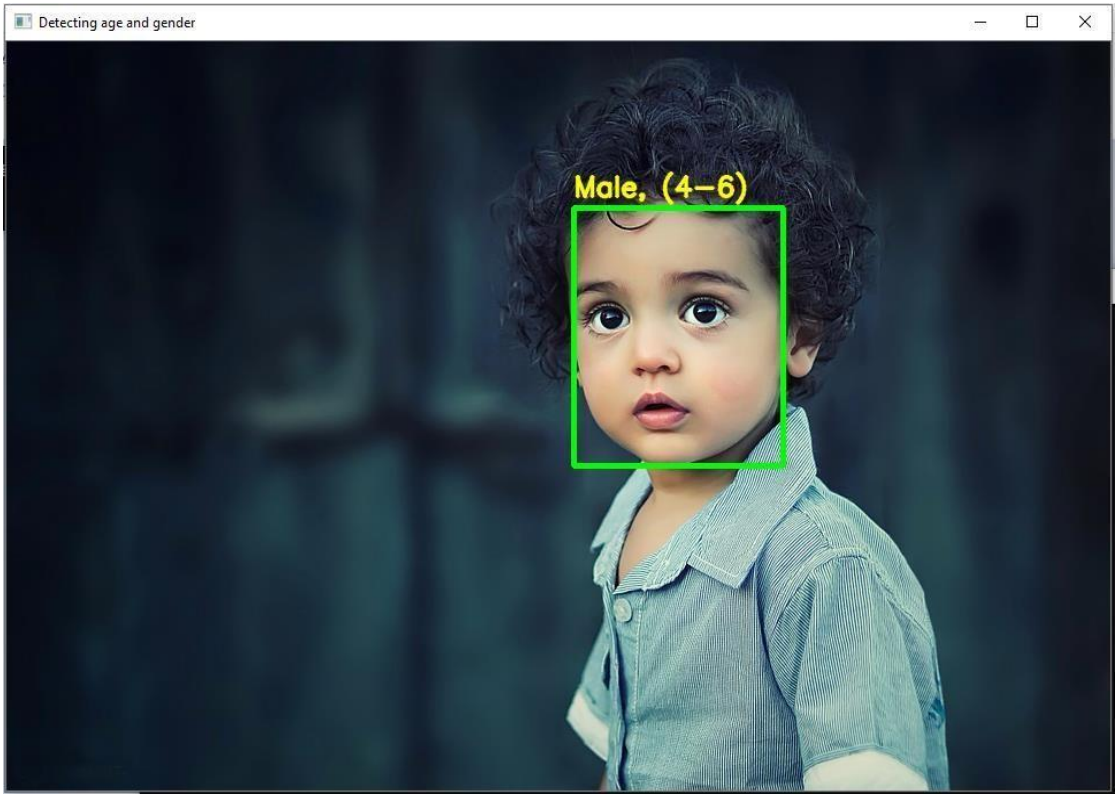
Age and gender estimation:

In general, age estimation can be characterized as a multi-class or regression problem, and gender estimation is defined as a two-class problem. After the face and eyes are detected, the facial image can be normalized as a fixed-size image using the localized eye positions. Then, the Gabor-LBP histogram framework (Gao et al., 2008) is used to extract the features for representation power of the spatial histogram (Lee et al., 2012). Using Gaussian derivative filters, the dimensions of the facial features can be reduced by 60% compared with the Gabor wavelet filters

5. Results

Though many previous methods have addressed the problems of age and gender classification, until recently, much of this work has focused on constrained images taken in lab settings. Such settings do not adequately reflect appearance variations common to the real-world images in social websites and online repositories. Internet images, however, are not simply more challenging: they are also abundant. The easy availability of huge image collections provides modern machine learning based systems with effectively endless training data, though this data is not always suitably labelled for supervised learning. Taking example from the related problem of face recognition we explore how well deep CNN perform on these tasks using Internet data. We provide results with a lean deep-learning architecture designed to avoid overfitting due to the limitation of limited labelled data. Our network is “shallow” compared to some of the recent network architectures, thereby reducing the number of its parameters and the chance for overfitting. We further inflate the size of the training data by artificially adding cropped versions of the images in our training set. The resulting system was tested on the Adience benchmark of unfiltered images and shown to significantly outperform recent state of the art. Two important conclusions can be made from our results. First, CNN can be used to provide improved age and gender classification results, even considering the much smaller size of contemporary unconstrained image sets labelled for age and gender. Second, the simplicity of our model implies that more elaborate systems using more training data may well be capable of substantially improving results beyond those reported here. We have proposed a SAM(supervised appearance model), which improves on the traditional AAM(active appearance model).. When used for facial feature extraction, the model describes the face with very few components. For instance, we usedjust 13 components to effectively represent the face on FGNET-AD as opposed to AAM, which requires between 50 and 200 parameters. When used for age estimation, we achieved 5.49 MAE,which is comparable to most state-of-the-art algorithms and better than most algorithms that used AAM for featureextraction.

6.Outputscreenshots



7. References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. Trans. Pattern Anal. Mach. Intell., 28(12):2037–2041, 2006.2

[2] S. Baluja and H. A. Rowley. Boosting sex identification performance. Int. J. Comput. Vision, 71(1):111–119, 2007.2

[3] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In Int. Conf. Mach. Learning, volume 3, pages 11–18, 2003.2

[4] W.-L. Chao, J.-Z. Liu, and J.-J. Ding. Facial age estimation based on label-sensitive learning and age-oriented regression. Pattern Recognition, 46(3):628–641, 2013. 1,2

[5] K.Chatfield,K.Simonyan,A.Vedaldi,andA.Zisserman.Returnofthedevilinthedetails: Delving deep into convolutional nets. arXiv preprint arXiv:1405.3531, 2014. 3

[6] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao. Wld: A robust local image descriptor. Trans. Pattern Anal. Mach. Intell., 32(9):1705–1720, 2010.2

[7] S. E. Choi, Y. J. Lee, S. J. Lee, K. R. Park, and J. Kim. Age estimation using a hierarchical classifier based on global and local facial features. Pattern Recognition,44(6):1262–1281,2011.2

[8] T.F.Cootes,G.J.Edwards,andC.J.Taylor.Activeappearancemodels.InEuropeanConf. Comput. Vision, pages 484–498. Springer, 1998. 2

[9] C. Cortes and V.Vapnik. Support-vector networks. Machine learning, 20(3):273–297,1995.2

[10] E. Eidinger, R. Enbar, and T. Hassner. Age and gender estimation of unfiltered faces. Trans. on Inform. Forensics and Security, 9(12), 2014. 1, 2, 5,6