SpeculAR

2

Introduction

“An Application that offers an intuitive real time virtual try-on solution for accessories like prescription eyewear and sunglasses.”

3

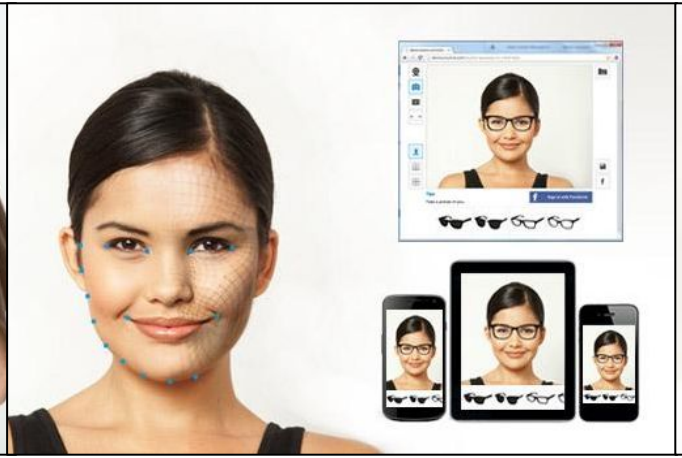
Inspired by...

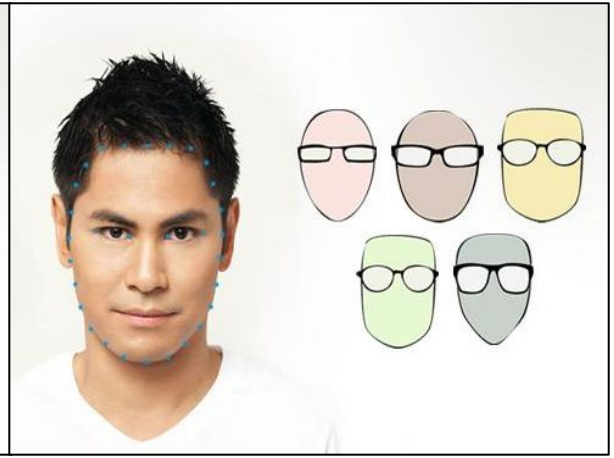
4

TryLive

AcepTryLive®

- A platform for retailers and e-commerce merchants to take their showrooms to the customer’s home.





5

Masquerade

Masquerade

- Allows recording video selfie animations and taking selfie photographs by changing the way you look.

6

Snapchat

snapchat

- Image messaging application software product that consists of a feature to create special graphical effects to user's face.

7

SpeculAR

8

Current status...

LiveRoom supports only “***environment augmentation***” through the back facing camera.

It does not have support for ***“self augmentation”*** through the front facing camera.

Environment Augmentation

Self Augmentation

9

Problem Description

Need

Placing virtual accessories realistically on the human face providing a seamless AR experience

Requirement

“An approach that allows real time tracking of face and estimation of head pose for mobile platforms”

10

Solution

“A real time face tracking and head pose estimation plugin,

that is optimized for the front facing camera in mobile devices,

and can be integrated with any Unity3D based application”

11

Requirement Specification

12

Functional Requirements

- **View the augmented model of the spectacles or accessory** on oneself on different perspectives.
- **Select a desired model**.
- **Update the models**.
- **Make an order**.
- **Take a snapshot or record a video** and share it on social networks.

13

Usability Requirements

- Easy Navigation
- Great Aesthetics
- Platform Usability
- Training time

14

Other Requirements

Supportability Requirements

- Should support its functionality in ⇒
  - **Different kinds of environments like indoor, outdoor** etc.
  - Under **different lighting conditions**.

Performance Requirements

- **Response time**
  - Head pose has to be calculated real time and the speed of interaction should be high.

15

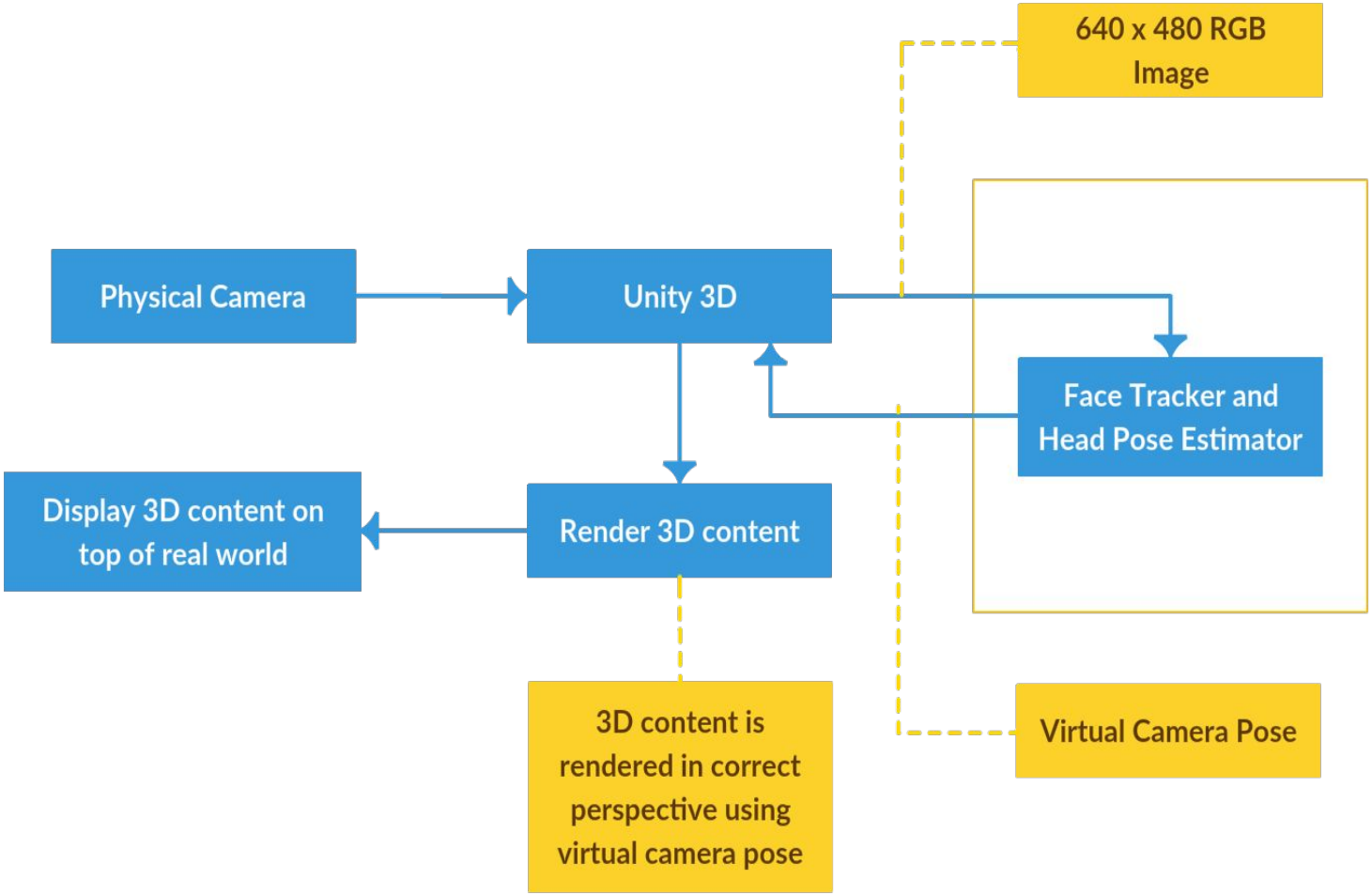
Design Constraints

- **Low processing power** and **memory limitations** in mobile devices.
- **Quality of the video stream** captured by the front facing camera of mobile phones being **relatively low**.
- **Different illumination conditions.**

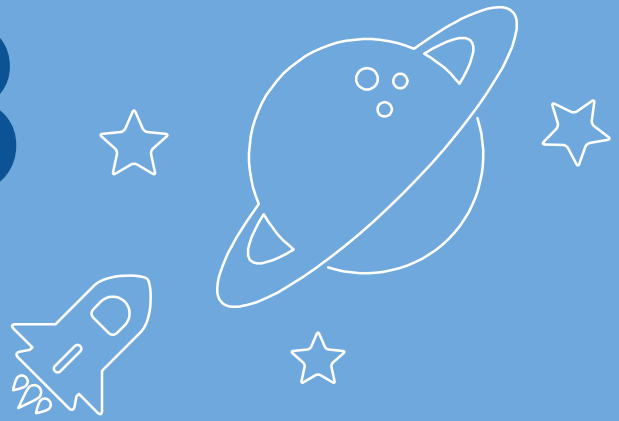
16

System Design

Architectural Representation



18



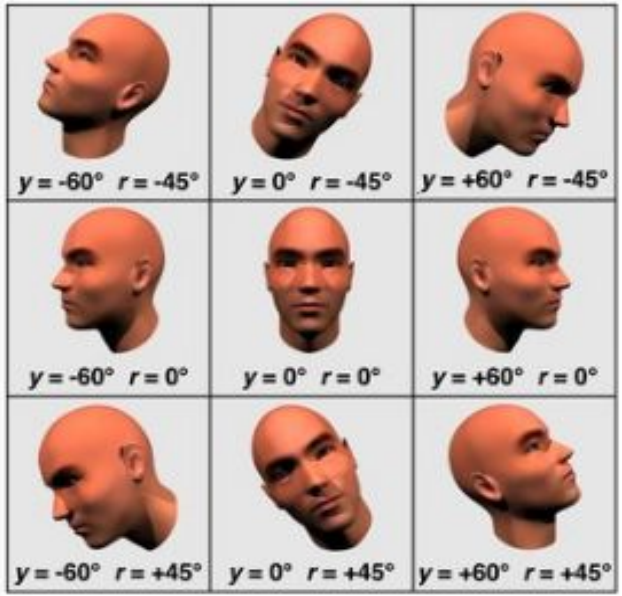
Existing APIs & SDKs

19

Google Mobile Vision API

Mobile Vision

- A way to facilitate finding objects in photos and video, using real-time on-device technology.

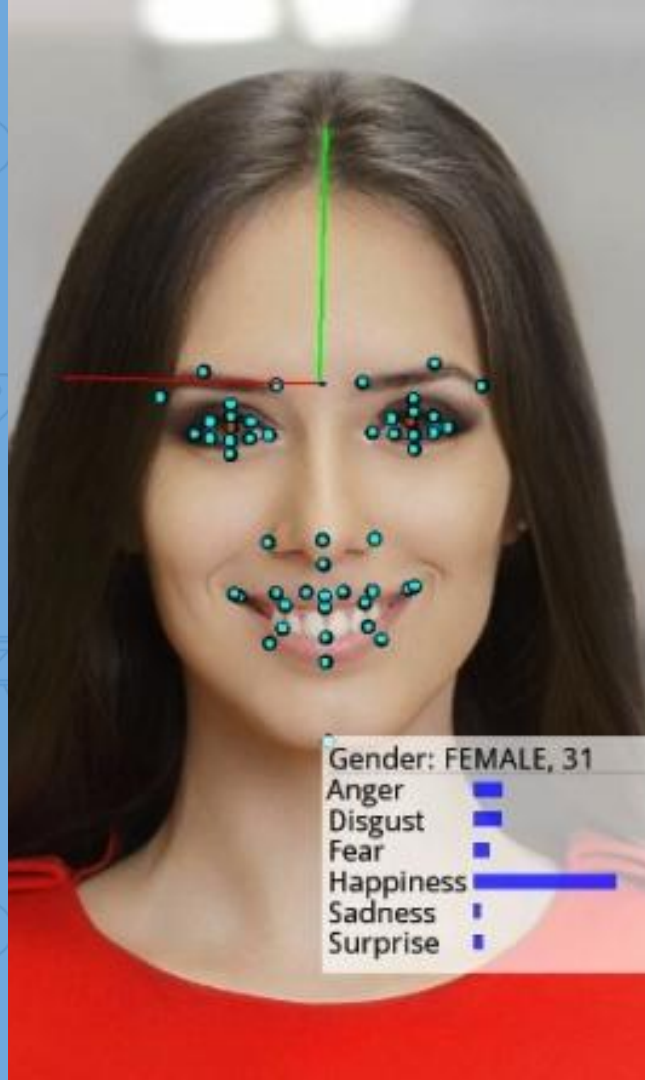


Example facial landmarks identified and tracked by Google Mobile Vision API

Pose angles determined by Google Mobile Vision API where y = yaw and r = roll

20

Visage Technologies visage|SDK™



- Could be utilized to develop applications that involve
  - **Interactive animation**
  - **Avatar control**
  - **Sophisticated user interaction**
  - **Advanced features in games, arts and other applications**
- Provides ***accurate tracking of head pose***, ***gaze direction***, ***facial actions*** and ***eye closure***.

21

OpenCV (Open Source Computer Vision Library)

- OpenCV library can be used for many applications such as ⇒
  - **Detecting and recognizing faces**
  - **Identifying objects**
  - **Tracking camera movements**
  - **Tracking moving objects** and many more.

22

Our research...

23

Methods

- Related peer reviewed journals and conference papers were searched mainly on the Google Scholar database.
- Additional conference papers and journal articles were found going through the reference sections.
- Websites which contained information on latest APIs, SDKs and commercial AR applications in the mobile AR domain were searched.

24

Workflow

25

Face Detection and Feature Extraction

26

Face Detection Approaches

- **Viola-Jones Object Detection Framework**

  Based on a cascade of weak classifiers that can detect separate Haar-like features identified in a facial image.

- **Using Eigenfaces**

  A template matching problem.

- **Skin based face detection**

  Using human skin as a mode of detecting a human face and its enclosing features.

- **Neural Networks based face detection**

  Utilizes a retinally associated neural network system to inspect small windows of a picture and choose whether each window contains a face.

27

Face Detection Approaches

- **Support Vector Machines (SVMs)**

  The hyperplane of the SVM is found by solving a linearly constrained quadratic programming problem.

- **Facial feature extraction**

  Identifying and locating interesting features in a facial image like eyes, eyebrows, nose and mouth etc.

- **Color based face detection**

  Skin color analysis.

- **Correlation**

  Finding the minimum error between a certain template and a patch in the image.

28

Face Tracking

- **Kanade-Lucas-Tomasi (KLT) tracking algorithm**

  Detects a set of object points across the video frames and then tracks the detected features between frames.

  KLT feature selection: Find the coordinates in the image which have a varying texture.

  KLT feature tracking: Select features to track based on texture (intensity information). Use Sum of Squared intensity Differences (SSD) as the measurement criterion to realize the tracking of feature points.

- **Active Shape Models (ASMs)**

  A statistical model of shape that captures the variability of a particular object class given an annotated set of training images

30

Face Tracking Approaches

- **Active Appearance Models (AAMs)**

  A way of matching a statistical model of shape and appearance to facial images. The model is built by combining a model of shape variation with a model of texture variation.

- **Constrained Local Models (CLMs)**

  Uses a joint shape and texture appearance model to generate a set of region template detectors.

31

Head pose estimation

32

Head Pose Estimation Algorithms

- **POSIT Algorithm**

  Pose from Orthography and Scaling with ITerations. For estimation of the 3D pose of an object from a 2D image.

- **Geometric Head Pose Estimation**

  Estimating the head pose or the gaze direction based on the geometric positioning of features extracted from a human face.

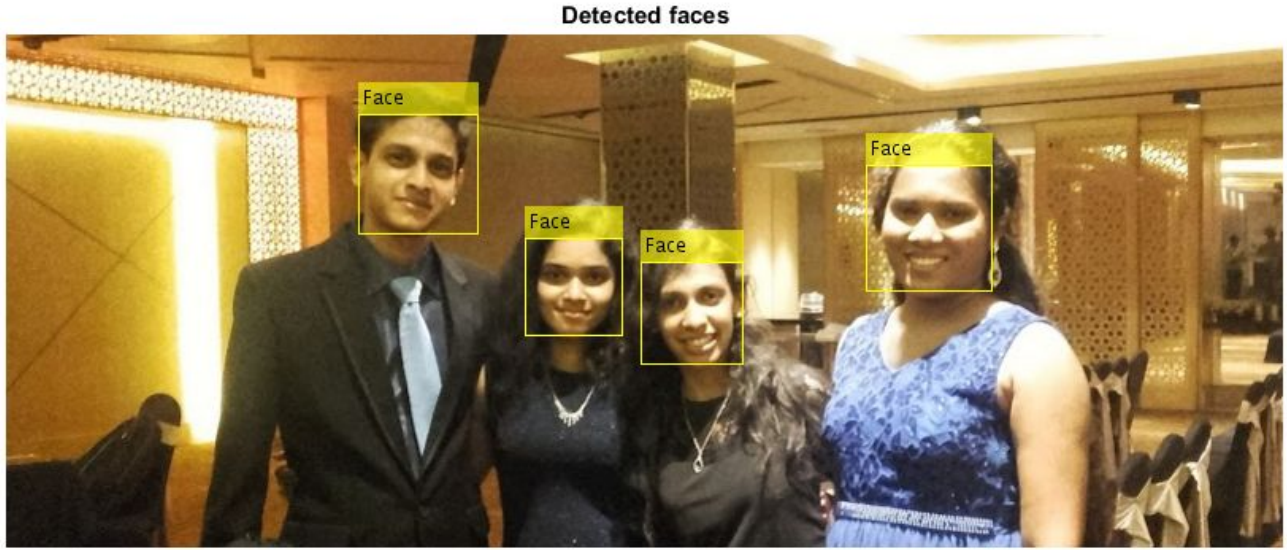
  Once the positions of the two eyes, two corners of the mouth and the nose tip is correctly found, the facial normal can be easily computed from them.

33

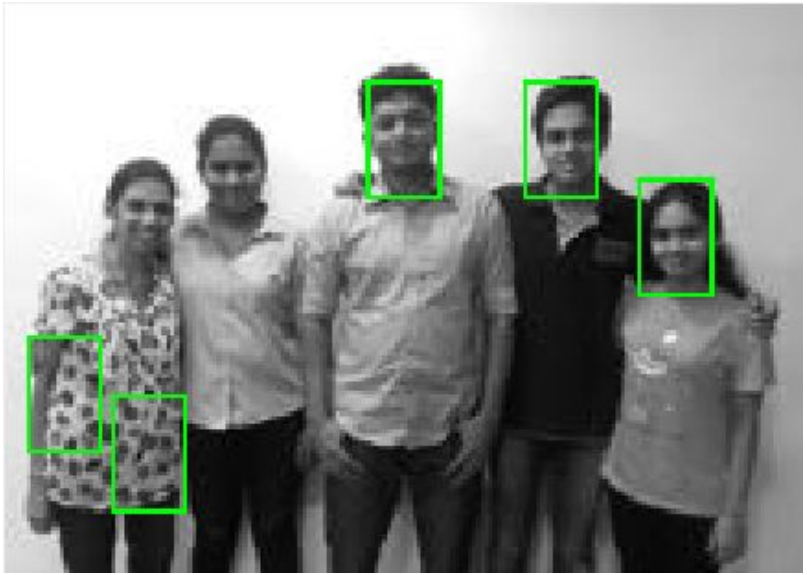
Implementations

Selected approach implementations on a PC platform.

- **Viola-Jones Face Detection**
  - Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
  - Elapsed time: 0.3025 sec.
  - Results:

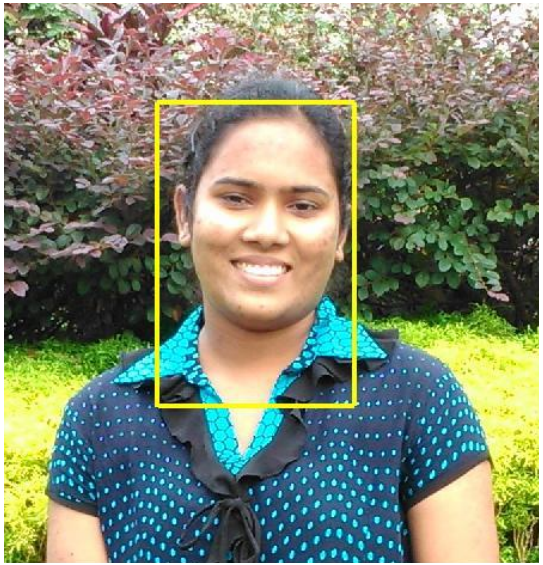


- **Neural Network based face detection**
  - Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
  - Elapsed time: 10.5820 sec.
  - Results:



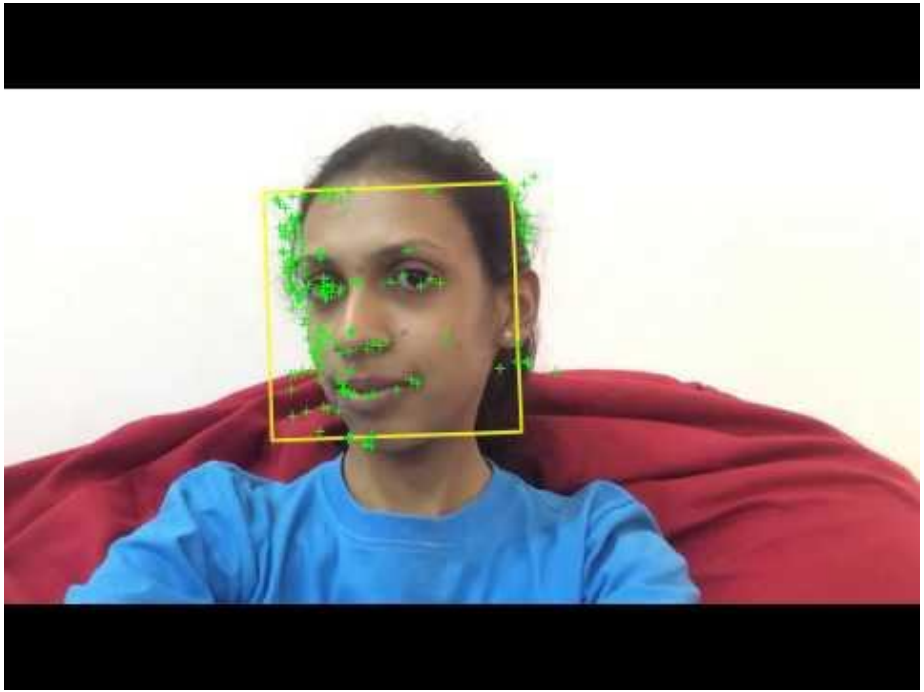
- **SVM based face detection**
  - Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
  - Elapsed time: 202.0102 sec.
  - Results:

- **Color Based Face Detection**
  - Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
  - Elapsed time: 3.0996 sec.
  - Results:



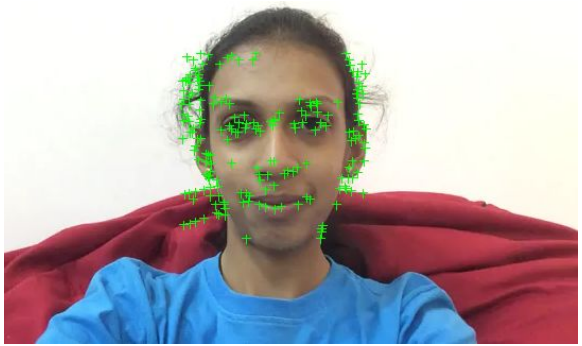
KLT Face Tracking

- Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
- Tracking speed: 27.4562 frames/sec.



RANSAC

- Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
- Tracking speed: 5.0121 frames/sec.

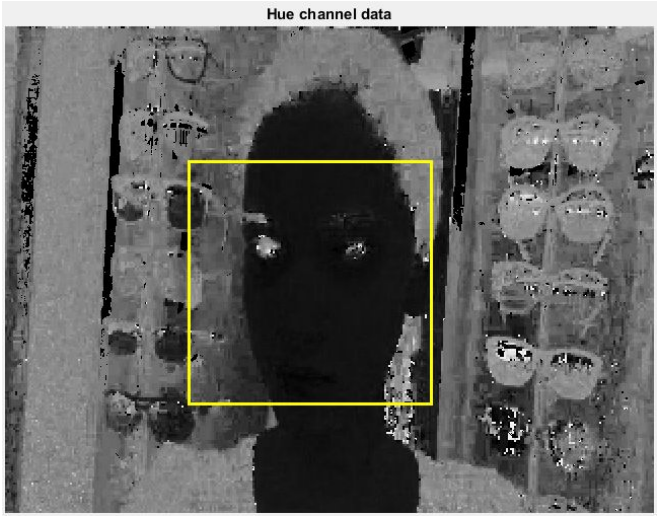


Detected features

Face Tracking Approaches

Camshift Tracking

- Tested Platform: PC Platform (Intel Core i5 1.80GHz CPU and a 4GB memory)
- Tracking speed: 26.4563 frames/sec.

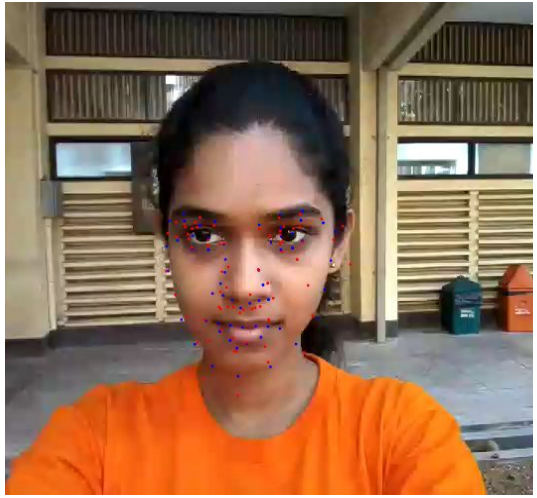


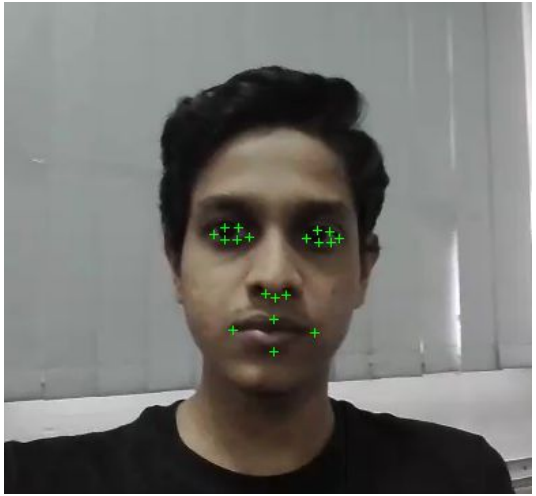
41

Facial Fitting Approaches

⇐ ASM

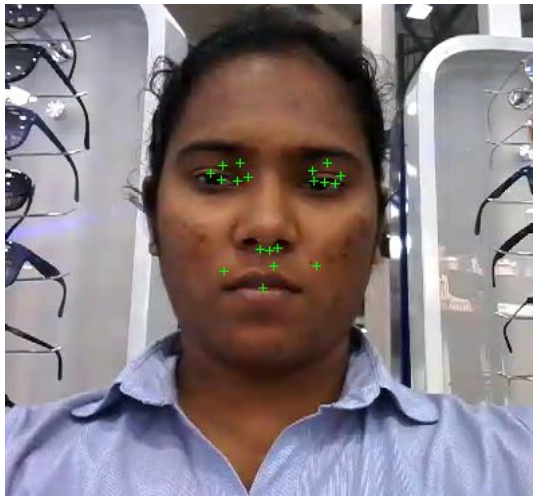
ASM ⇒





⇐ AAM

CLM ⇒



42

Head Pose Estimation Approaches

- **Geometric Pose Estimation**
  - Elapsed time: 2.7095×10^{-4} sec.
    - To derive the pitch, yaw and roll angles from the 5 facial features detected in a single frame.
- **POSIT algorithm**
  - Elapsed time: 0.0087 sec.
    - To derive rotation and translation by matching 19 2D model points into the 3D model

43

Test Video Suite

Test Videos under different environmental conditions, devices and camera specifications.

44

Test Video Suite

Environment: Indoor environment

Lighting condition: Diffused light

Device Specifications: Samsung Galaxy Grand Prime

Camera Specifications:

5 MP, f/2.2, 1080p

Environment: Outdoor environment

Lighting condition: Diffused light

Device Specifications:

Samsung Galaxy Core Prime

Camera Specifications:

2 MP

45

Test Video Suite ctd.

Environment: Living room

Lighting condition: Diffused light

Device Specifications: iPhone 6

Camera Specifications: 1.2 MP. f/2.2, 720p video

Environment: Opticians Showroom

Lighting condition: Shop lights

Device Specifications: Samsung Galaxy Grand Prime

Camera Specifications:

5 MP, f/2.2, 1080p

46

Viola-Jones Face & Feature Detection + KLT + Geometric Head Pose Estimation

20.8966 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory

47

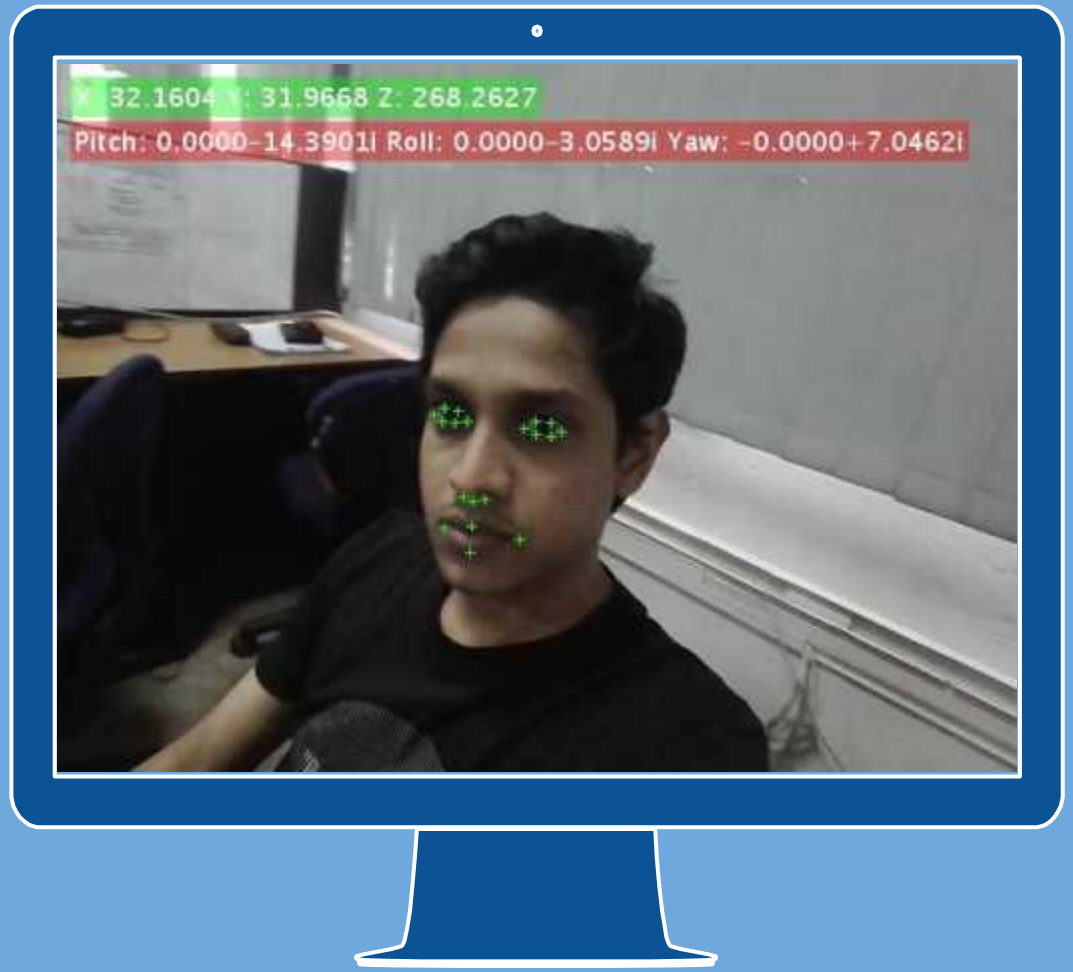
Viola-Jones Face Detection + AAM + POSIT Head Pose Estimation

7.2326 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory

48

Viola-Jones Face Detection + Lucas-Kanade Initialization + AAM + POSIT Head Pose Estimation

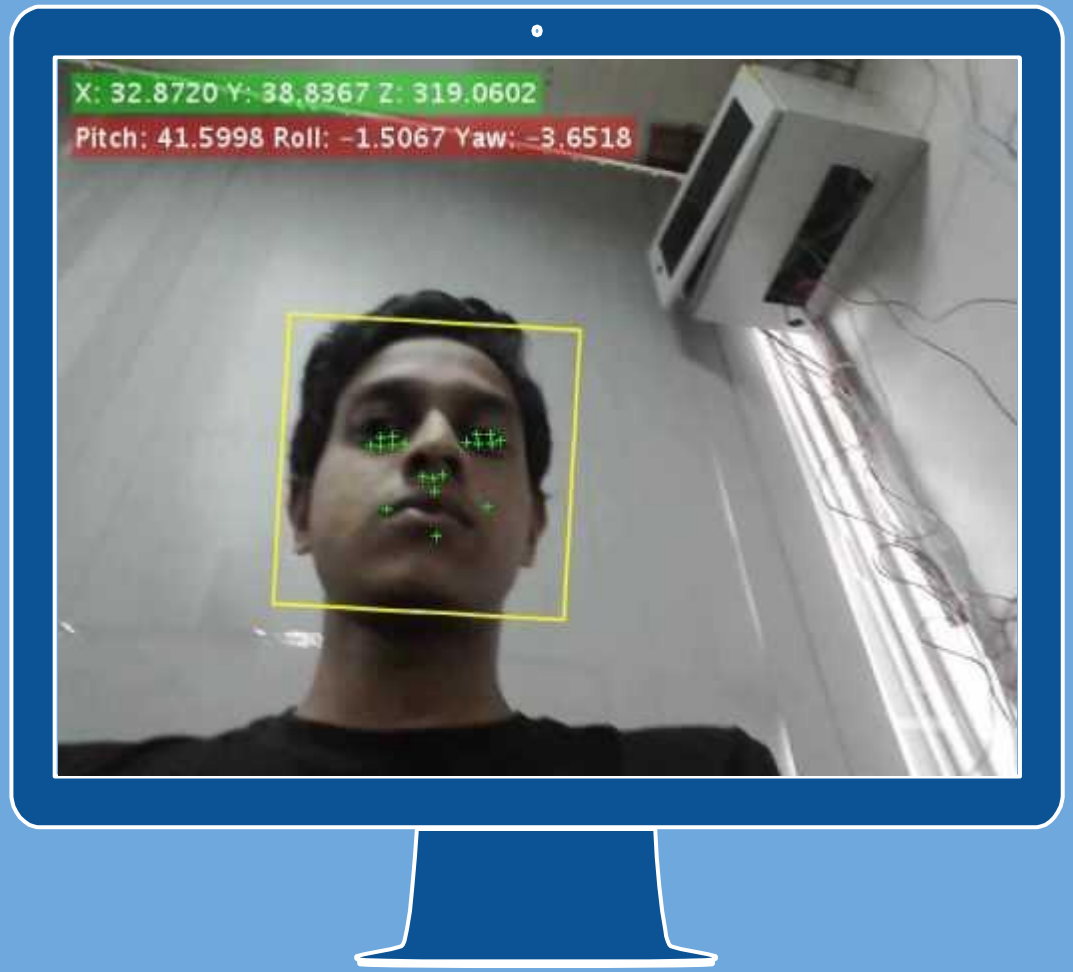
4.5672 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory



49

Viola-Jones Face Detection + Lucas-Kanade Initialization + AAM + Divergence Check + POSIT Head Pose Estimation

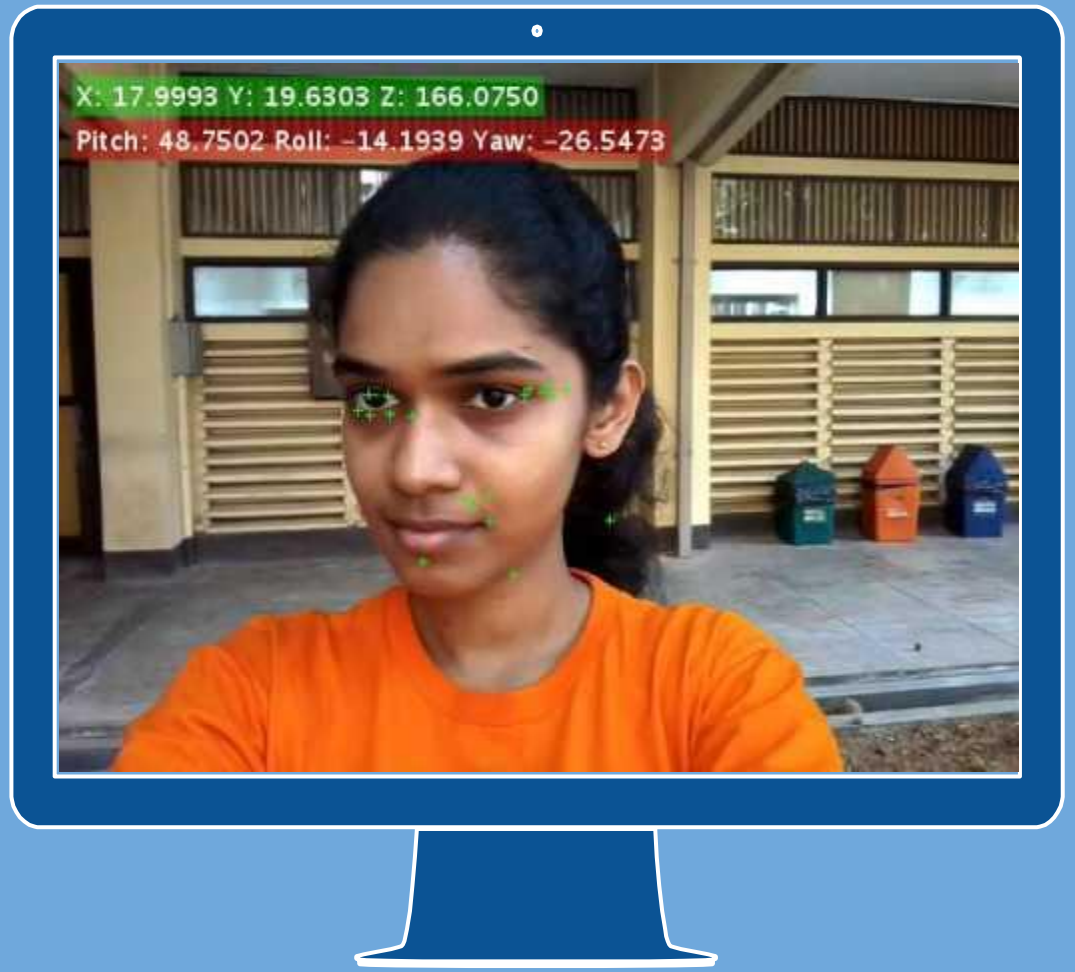
4.5654 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory



50

Viola-Jones Face Detection + CLM + POSIT Head Pose Estimation

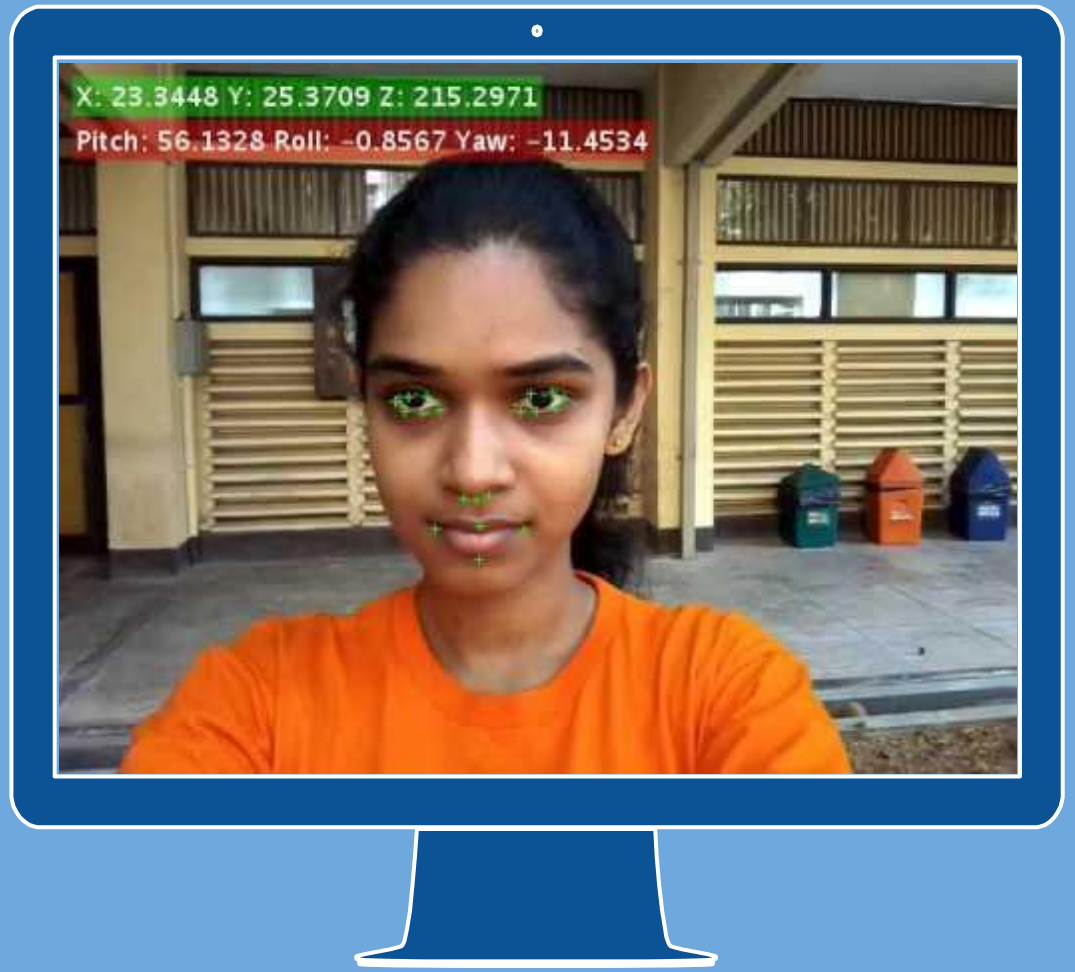
0.7158 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory



51

Viola-Jones Face Detection + Lucas-Kanade Initialization + CLM + POSIT Head Pose Estimation

0.4563 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory

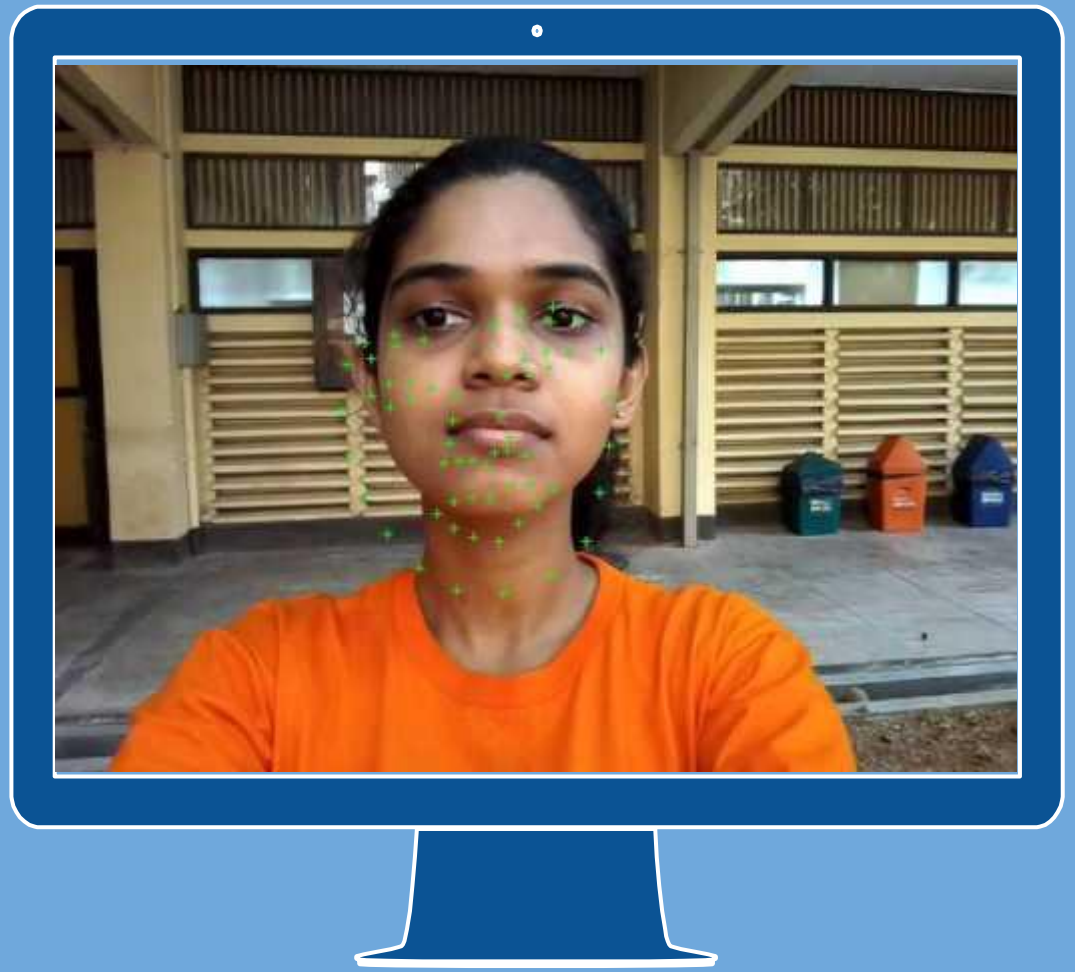


52

Viola-Jones Face Detection + ASM + POSIT Head Pose Estimation

Trained on frames with significant pose variation of test video.

0.2832 frames/sec. on a PC platform with Intel Core i5 1.80GHz CPU and a 4GB memory



THANKS!