## Features:

* Accesses webcam
* Streams live video from webcam
* Takes image input at runtime
* Determines the user’s face using a resizable oval
* Asks the user to adjust eyes and the mouth using movable ovals
* Determines all the coordinates of the face, eyes and mouth using equation of ellipse
* Keeps resource images with known values of eyes, face and mouth
* Accesses the resource image and morphs the two images
* Displays the morphed image

## Algorithm:

### Input:

The first frame that appears has three buttons which allow the user to either start a new frame or go for a tutorial or exit the application.

When the new button is clicked, the gender of user is first asked. Then, video is streamed accessing the webcam using openCV library of java. An oval appears on the video .The user is supposed to adjust his face such that it fits inside the oval. When the user has adjusted accordingly, he may capture the image.

### Adjusting Components:

After that image is displayed on screen along with movable ovals which the user adjusts such that it overlaps with the boundary of user’s face, eyes and mouth. When the user has made all the adjustments, he clicks on the done button. Then, the coordinates of all the ovals are determined using equation of ellipse.

### Resource Images:

The application also contains image resources that include pictures of other people’s faces. We already know the position and coordinates for face, eyes and mouth of these pictures. The image is morphed in such a way that eyes and mouth of the user is mapped on to the eyes and mouth of the image stored.

### Morphing of Eyes and Mouth:

This is done using arrays and initializing the arrays of both the image stored and the image captured. When the point is reached where value of array for eye of stored image is 1, it looks for value 1 in captured image. Then it sets the RGB of the stored image at that point equal to the RGB of value 1 for captured image. It loops through all the coordinates and checks for corresponding coordinates in the captured image.

### Morphing of Face:

The values for face are average of the two images. This is done by getting the pixel value for both the images, shifting to separate the red, blue and green values. Then average of the two reds is taken. Similarly average of the two greens and blues is taken. The RGB is then combined using the OR operation. The value for that particular pixel is then set to the new average value.

## Classes:

### Project.java

It sets the layout of the first frame of the application, contains buttons to initiate the video streaming using webcam. It implements the MouseListener and the KeyListener interface.. The animation on the frame is drawn using the paintComponent method.

### Oval.java

It creates interactive oval shapes to recognize where the face, eyes and mouth is present. By clicking the move button, the oval for eyes can be moved. Therefore, it allows the user to adjust it such that it marks the boundary of user’s face, eyes and mouth.

Then the arrays for the coordinates of these components are initialized in in a method called cord.   
Its morph method creates buffered images of the two images to be morphed. It then sets each pixel value to the morph the two images.

### Mat2Image1.java

This is the class that deals with video streaming and image capturing.

## Control.java

This is the controller class. It is public and it contains the main method.