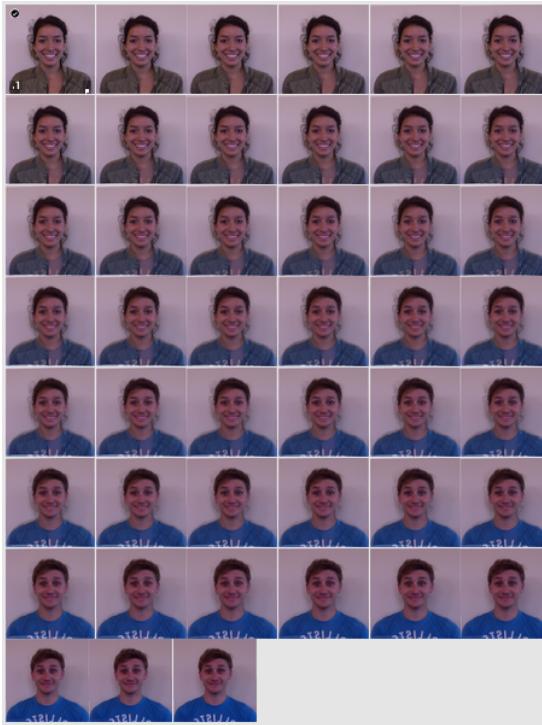


# Feature-Based Image Metamorphosis



Josh Israel  
Peng Hui How  
Simon Ayzman

# Objective

Morph two given images

||

|| (Possible to then...)

\\

Morph two given videos

# Procedure for Image Morphing

Morphing between two images:

- 1) Define the common features between the source and destination images
- 2) Create intermediate frames
- 3) Generate video from all of the intermediate frames

# **Procedures for Video Morphing**

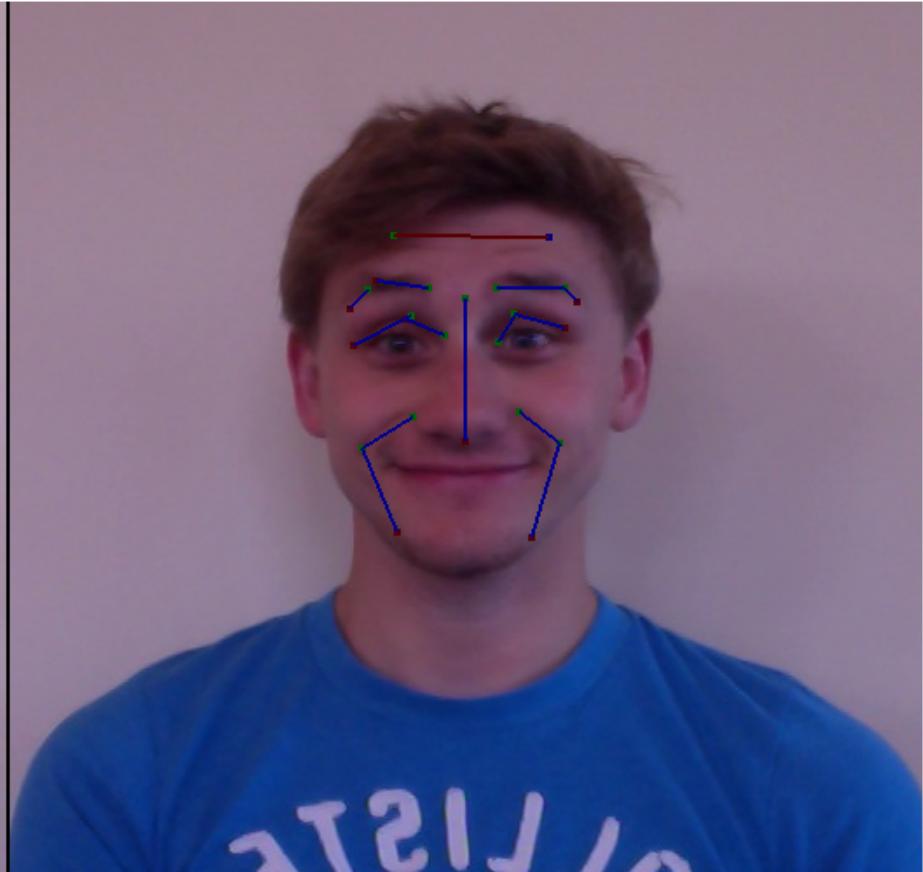
Morphing between two videos:

- 1) Split each videos into its respective frames
- 2) Create intermediate frames between each of the corresponding frames in the videos
- 3) Select appropriately interpolated/morphed frames for each frame in the final video
- 4) Generate video from chosen frames

# Programs

- 1) featureAdder
- 2) imageMorph
- 3) combiner
- 4) frameExtractor
- 5) *autoFeatures (!)*

# Feature Adder



# Feature Adder



# Feature Adder



# Feature Adder

A user interface that allows users to specify common features in both the source and destination images.

Other functions include: (1) undoing mistakes, (2) opening/editing previously made feature files, (3) taking screenshots

# Feature Adder

Input: Two images (source and destination),  
with other relevant parameters

Output: Two .feat files (source and destination)

# Image Morph



# Image Morph

Generates the intermediate frames between source and destination images and a video made up of the sequential intermediate images to simulate metamorphosis

Various easing functions are available

# Image Morph

Input: Two feature files, with other relevant parameters

Output: A video that demonstrates the metamorphosis between the 2 input images, and sequence of intermediate images that indicates the between them

# Combiner / Frame Extractor



# Combiner / Frame Extractor

(use *SimpleVideo* class!)

Video reader: Given an AVI video file, this class can produce a set of frames

Video writer: Given a sequence of PNG image files, this class can produce an AVI video file

# Frame Extractor

Input: An AVI video file

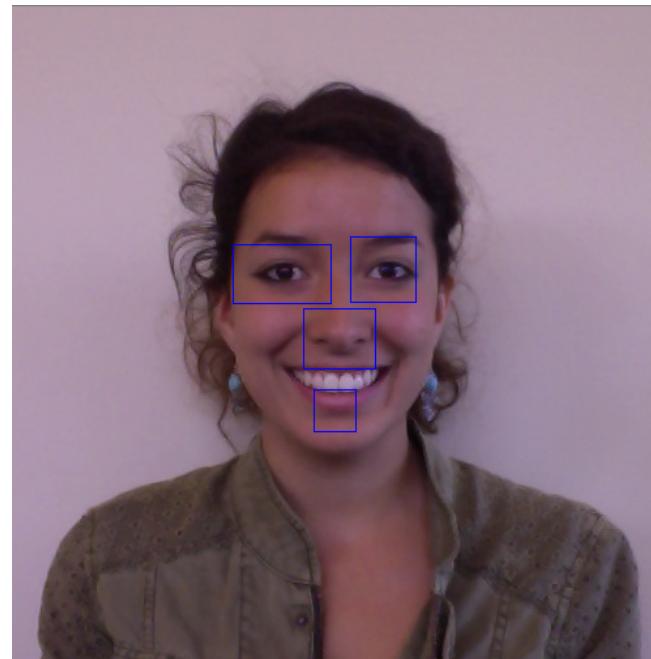
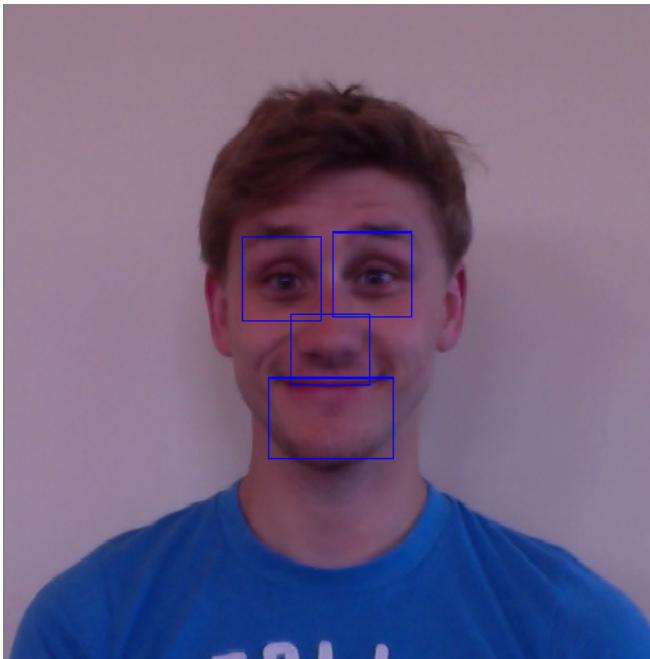
Output:  $n$  frames of the AVI video file (as PNG image files), for processing by Feature Adder

# Auto Features

Input: A PNG image file

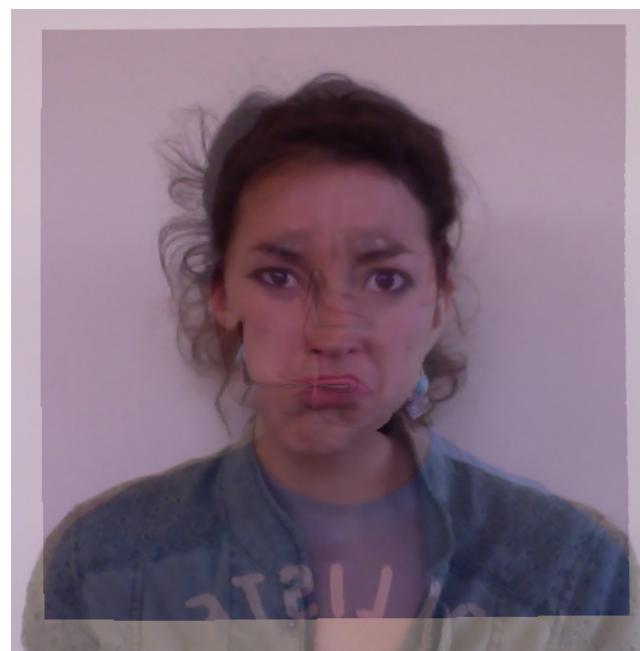
Output: A .feat file in the same format as Feature Adder, using facial feature detection with haar cascades (functionality from OpenCV). Each rectangle detected is turned into two features, by taking the top and left sides.

# Auto Feature



# Auto Feature

Poor results. Features are unreliable and inconsistent



# References

Feature-Based Image Metamorphosis, SIGGRAPH 1992,  
Thaddeus Beier.