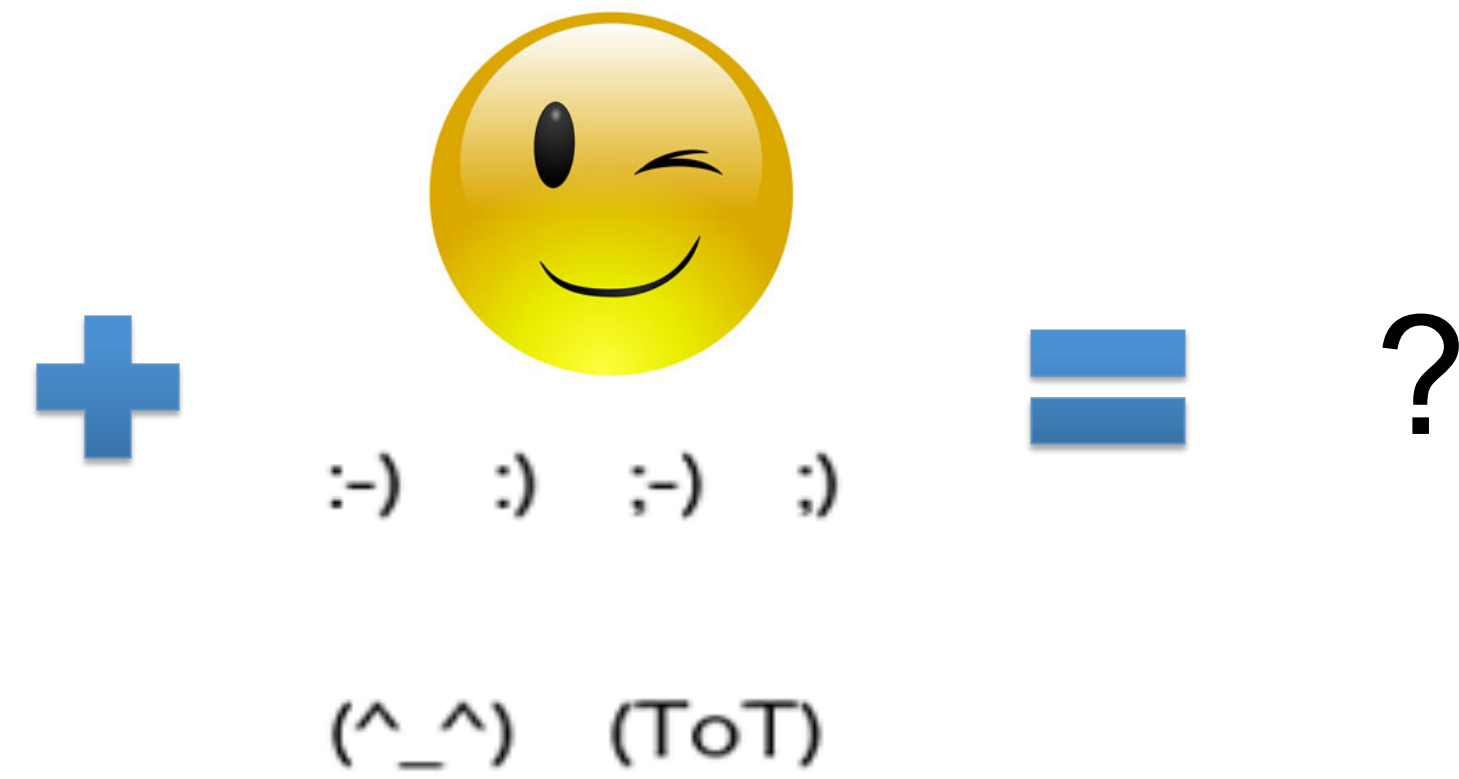
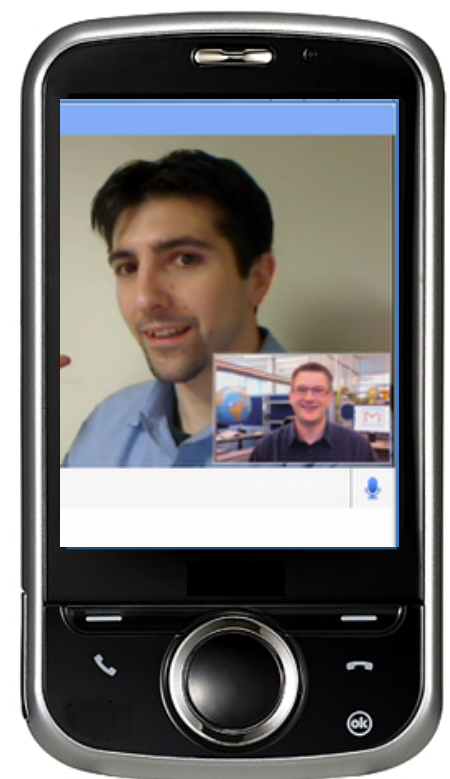


# Emoticonizer: Facial Feature Detection and Changing on Android

Kun Yi

Department of Electrical Engineering, Stanford University

## Motivation: Emoticons on Real Face!



### Goal:

Fast and robust facial feature changing to visualize emoticon

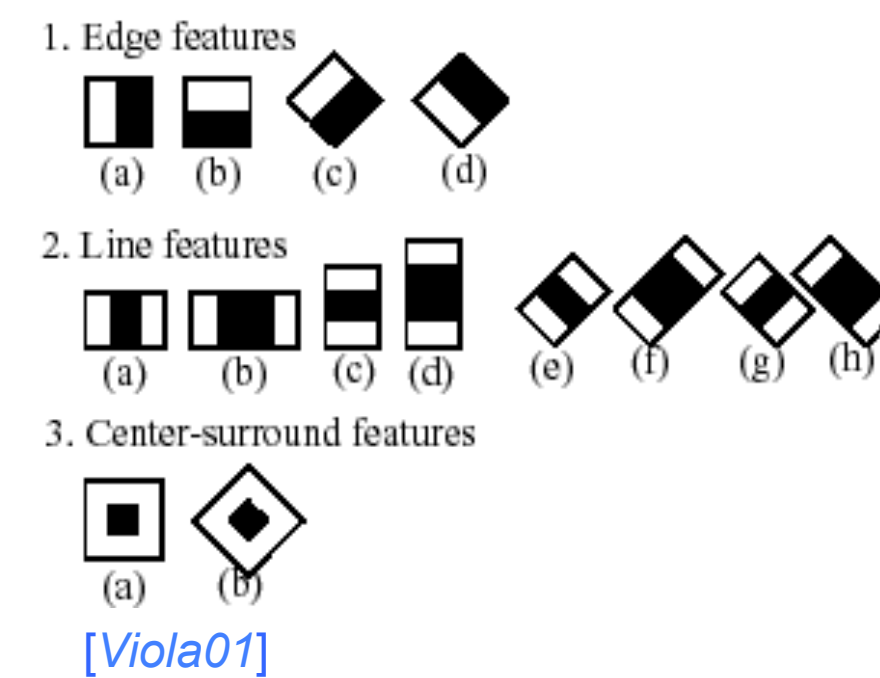
### Challenge:

Face feature replacement, stable tracking and performance

## Algorithms / Tools Used

### Face and Facial Feature Detection

- Used Haar Cascade Classifiers to detect face as well as facial features



- Fast, stable tracking of face

### Fast Real-Time Feature Tracking

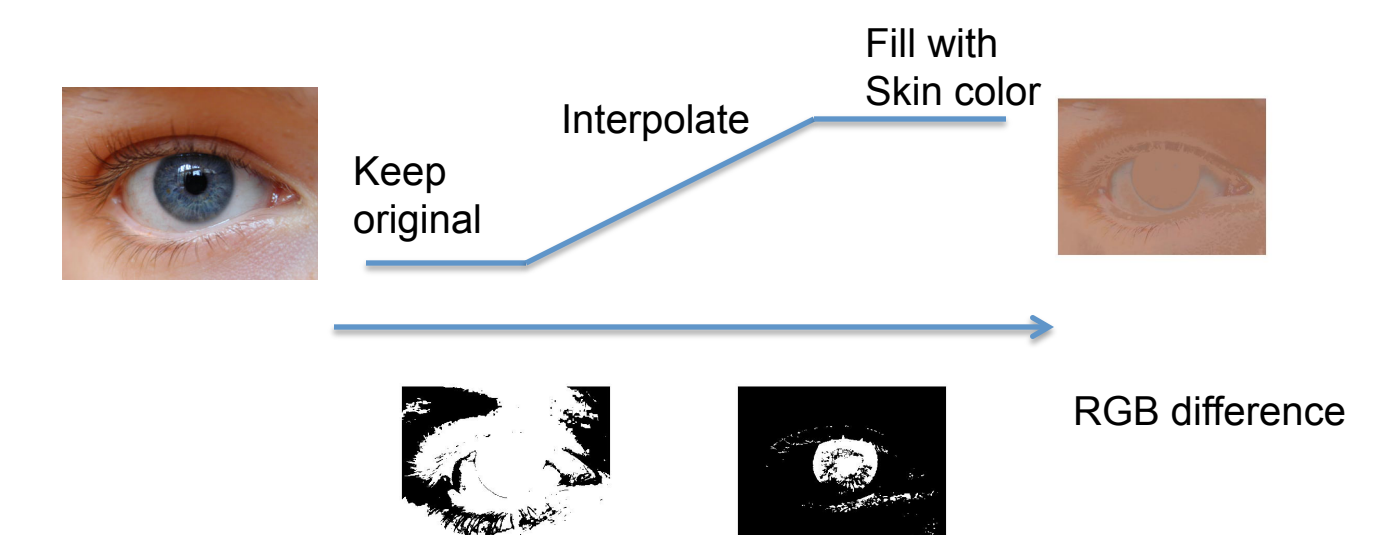
- Face detection more stable than feature detection
- Use face detection as reference to make up for "lost" features in a frame

```
if (abs(center(feature) - center(old)) > threshold)
{
    delta = center(face) - center(face_old)
    center(feature) = center(old) + delta
}
```

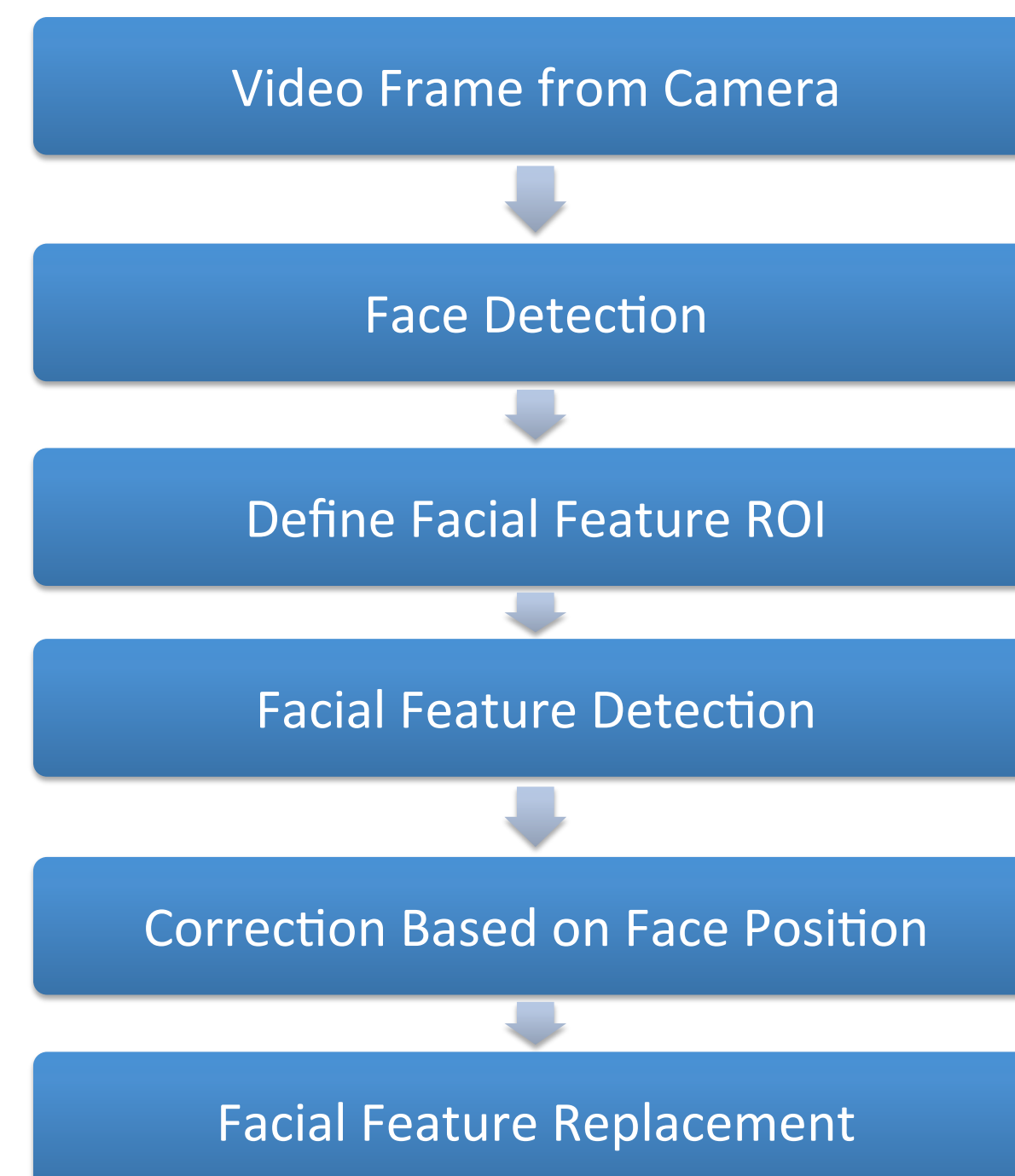
- Improves robustness

### Facial Feature Replacement

- Before drawing emoticons, need to remove feature and fill with skin color
- Blend pixels close to skin color to avoid filling a "block" of solid color
- Gaussian blurring for smoother result



## Workflow



## Results

### Device:

- Motorola Atrix
- 1GHz Dual Core
- 2G Memory
- 500 MP Camera
- Android 2.3.6

### Performance:

- Features stably detected and tracked
- Running speed reasonable

## Future Work

- Improve Blending Visual Effect
- Camshift Algorithm for Tracking
- Simple Emotion Detection

## References

[Viola01] Paul Viola and Michael J. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE CVPR, 2001. <http://www.ai.mit.edu/people/viola/>

Face Features Detection with OpenCV. <http://xuvif.blogspot.com/2011/05/face-features-detection-with-opencv.html>