COLOR COURSE PROJECT

Automatic Gaze Correction

Daniel Heilper
Moran Lefler

PROBLEM DEFINITION

- When photographing people, Their gaze (the direction the people look to) is not always directing the camera, resulting in a less-emotional impact.
- This is especially apparent in video conferencing, where the subjects cannot maintain eye-contact

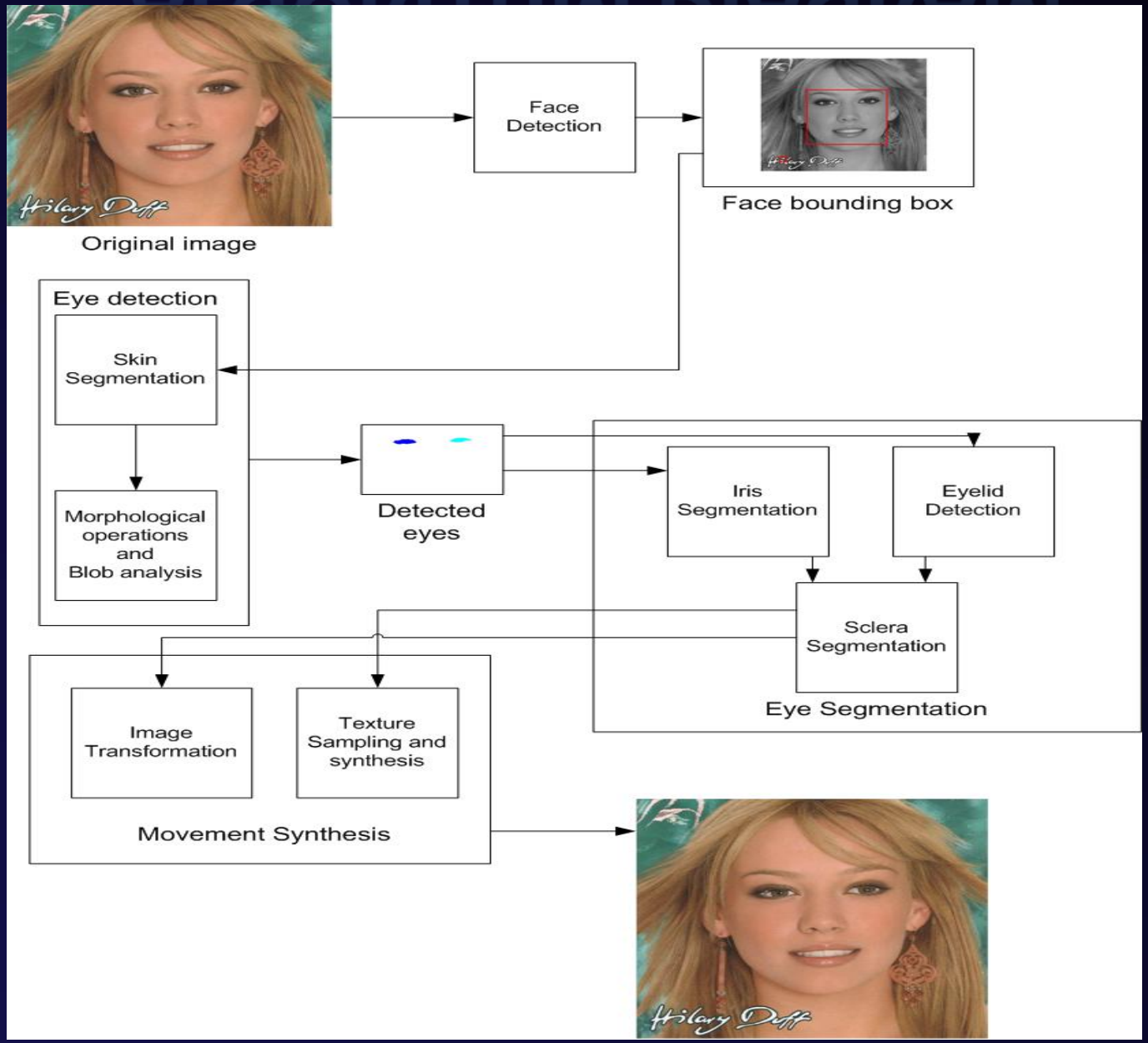
CURRENT WORK

- We have put our emphasis so far on eye movement synthesis.
- We have found it to be a non-trivial process, which includes many vision and color related tasks.
- Work description:
  - Input: image of a person
  - Output: image of person with corrected eyes, after movement

PROCESS DESCRIPTION

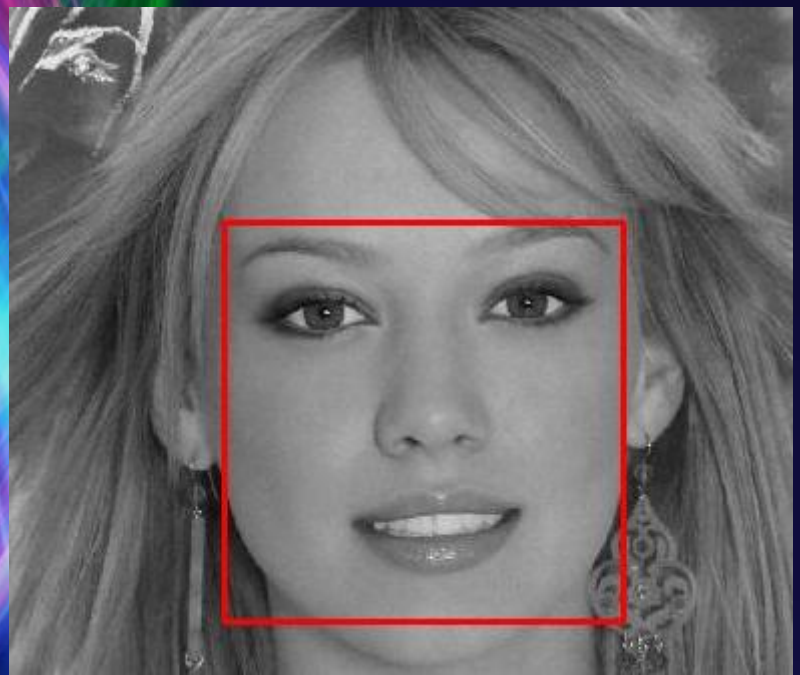
- Given an image:
  1. Detect faces in the image
  2. For each face:
  3. Detect eyes location
  4. Segment eye and retrieve measurements
  5. Compute transformation from given correction angle
  6. Apply transformation
  7. Fill missing areas

ALGORITHM DIAGRAM



DETECT FACES IN IMAGE

- Given an image of a person:
  1. We used Robust Real-time Object Detection by Viola & Jones to get bounding box around faces in the image.



at this stage, due to false alarm problems, we only use images with one person. The process still helps to crop the region of the face.

DETECT EYES IN FACE

- Given an image of a person's face (cropped from full image):
  1. We use a skin detection algorithm to segment skin areas.
     - The algorithm uses the hue Dimension from the HSV color space and transformation to YCbCr in order to map skin regions.

skin_mask =

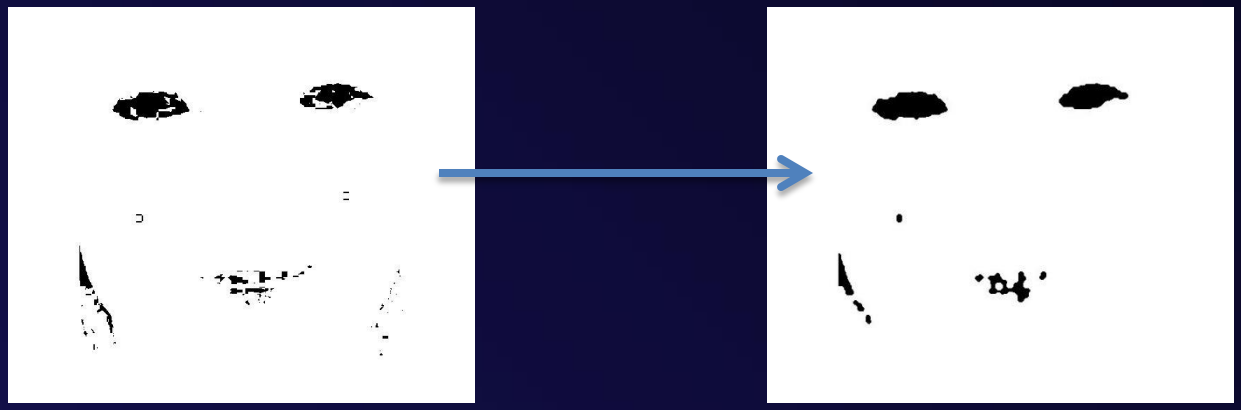
140 <= cb <=195 & 140 <= cr <=165 & 0.01 <= hue <= 0.1;

DETECT EYES IN FACE

- Skin segmentation based on *"Automatic Red-Eye Removal based on Sclera and Skin Tone Detection* by *Flavien Volken, Johann Terrier, Patrick Vandewalle"*
- *Cai and Goshtasby, J. Cai, A. Goshtasby, and C. Yu, "Detecting Human Faces in Color Images, 1998 Int'l Workshop on Multi-Media Database Management Systems."*

DETECT EYES IN FACE

1. Non-skin areas have the potential to be eye segments.
   - This means that this algorithm will only work on color images and will not work on intensity only images.
2. We use morphologic operations to avoid detecting non-skin regions which are not eyes.



DETECT EYES IN FACE

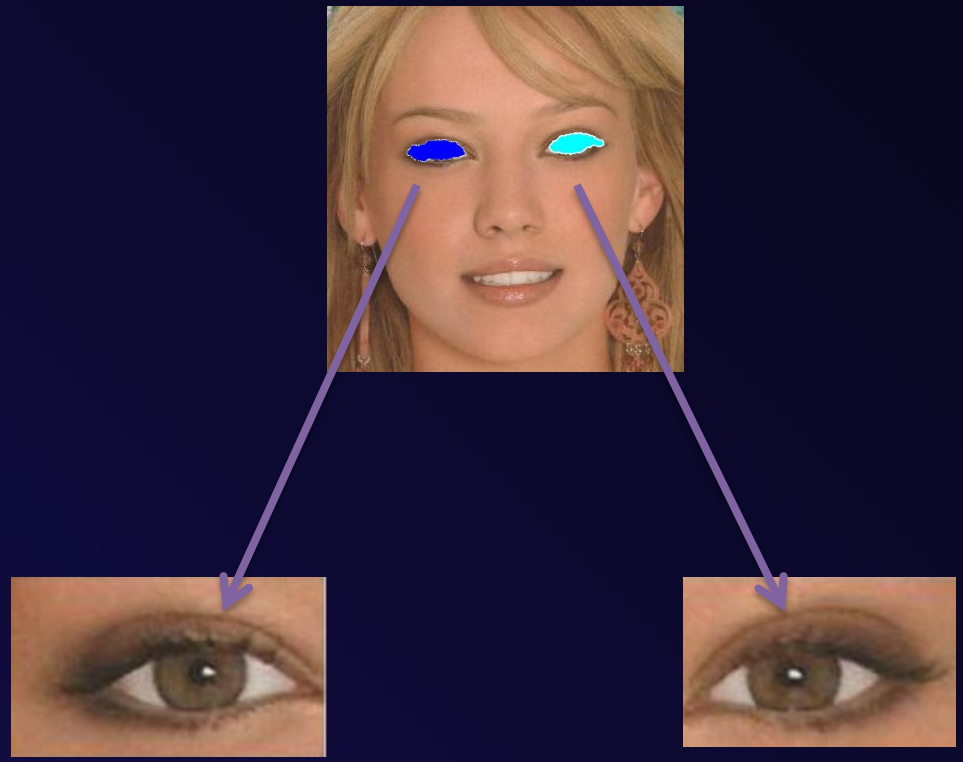
3. We then use a set of blob analysis properties to see whether each candidate blob behaves like an eye segment. The examined properties are:
   - Height - Eyes should be at the top half of the face
   - Solidity — Scalar specifying the proportion of the pixels in the convex hull that are also in the region. We expect a solidity value of more than 0.7
   - Area – we check that the area of the blob is large enough, and that the ratio between the blob area and the rest of the image is not too small, in order to exclude small non-skin regions.

DETECT EYES IN FACE

- (properties examination continued)
  - Hue variance – since the color disparity is large within the eye, we assume that the hue variance should be large.
  - Orientation– eye blobs should be larger in the horizontal axis than in the vertical axis.
  - The blob analysis assume that the face detection step was successful and that the face region was detected tightly around the face.

- NOTE: This does not result in accurate eye segmentation, but gives a good estimation for the location of the eyes.

DETECT EYES EXAMPLE

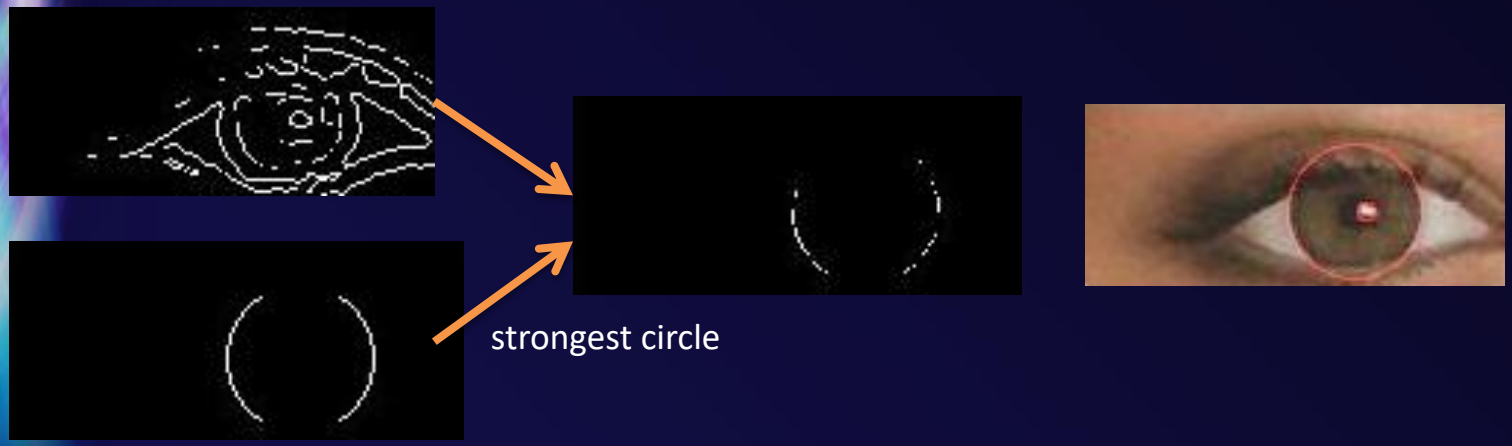


EYE SEGMENTATION

- For a successful eye correction process, the eye should be segmented as accurately as possible.
- In the eye segmentation step, we segment the different parts of the eye
  - Iris
  - Sclera
  - Eyelids

IRIS SEGMENTATION

- We have tested two different methods for iris segmentation.
  - Using Hough transform for circle detection. For each possible radius, we search for the maximum intersection between the edge map and the strongest circle obtained by the transform.



IRIS SEGMENTATION

- Other similar Iris segmentation concepts:

  John G. Daugman. High confidence visual recognition of persons by a teste of statisticalin dependence. In IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15,no. 11, pages 1148–1161, U.S.A., 1993

- Camus, T.A., and Wildes, R.: 'Reliable and fast eye finding in closeup images'. IEEE 16th Int. Conf. on Pattern Recognition, Quebec,Canada, 2004, pp. 389–394

EYELIDS SEGMENTATION

1. Find edges in the image using canny edge detector.

2. Find the topmost, leftmost, rightmost and bottommost points in the edge map.
3. Estimate 2 parabolas that will model the shape of the eye using ‘polyval’ in MATLAB on the found points

4. Every pixel between the two parabolas is marked as an eye pixel.

IMAGE TRANSFORMATION

- We find that for smaller angles, translation obtains satisfactory results.
- We work on the cropped bounding box of each eye separately.
- First we apply the eyelids mask on the original eye image

IMAGE TRANSFORMATION

- Then, we apply the transformation on the segmented image

- We then cut the original eye region from the translated image

TRANSFORMATION RESULTS

- Eye image after translation of the segmented area.

- The black region needs to be filled with missing sclera and iris parts

FILLING MISSING AREAS

- Two distinctive regions needs filling
  - Sclera (eye whiteness)
  - Iris (eye color)
- We determine the 2 regions using set operations on the image pixels
- The region are filled with texture synthesis method based on a sample from the available texture.
- We use texture synthesis based on:
  Image Quilting for Texture Synthesis and Transfer by *Alexei A. Efros and William T. Freeman from Proceedings of SIGGRAPH '01, Los Angeles, California, August, 2001*.

TEXTURE SYNTHESIS EXAMPLE

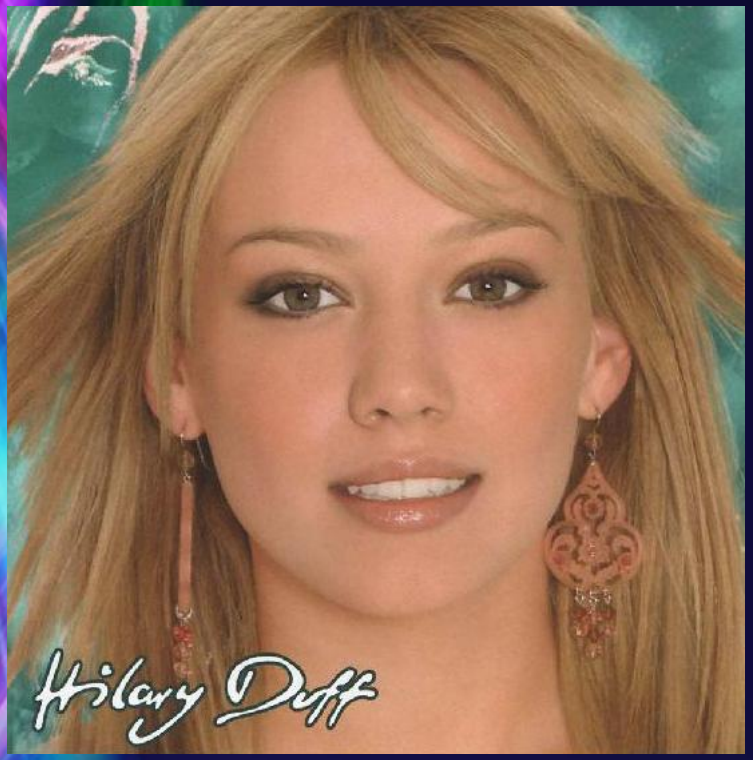
Eye after transformation

Eye after sclera and iris synthesis

OUTPUT RESULTS

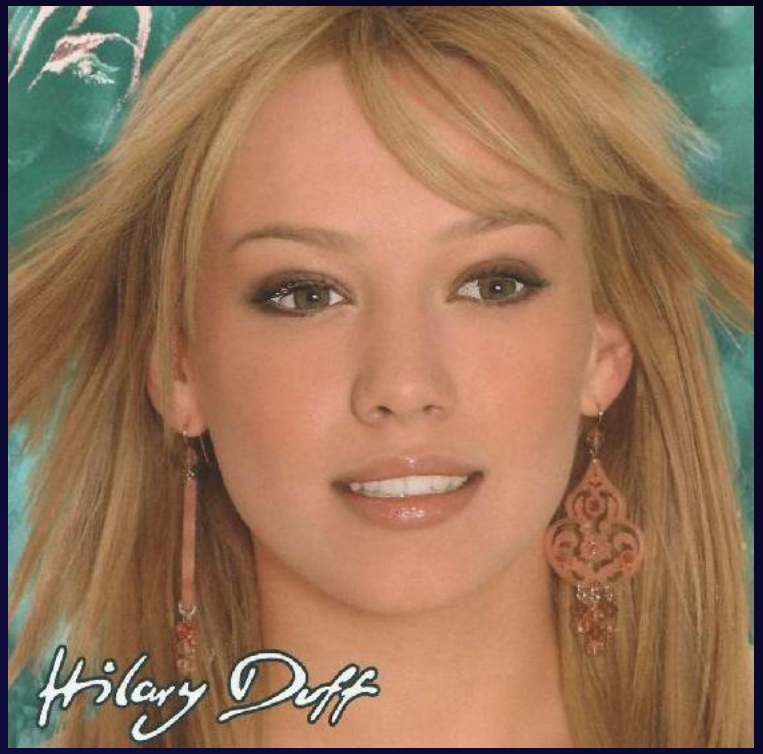
We then paste the transformed eye images back to the original image

Before



After

After



FUTURE WORK

- Correction angle computation
- Better Segmentation of eyelids
- Better transformation using 3-d eyeball model