We plan to implement face swapping on the Android phone. This will consist of a number of steps, ideally including:

1. Face detection.
2. Face outlining and extraction.
3. Choosing two or more faces to swap around.
4. Translation of each face into the proper positions.
5. 2D Rotation of each face into the proper orientations.
6. 3D Rotation of each face toward or away from the camera.
   a. This will require some face feature detection and interpolation
7. Stitching the faces onto their new bodies and smoothing the edges.
8. Lighting adjustment for more realistic effect.

Note: Steps 6 and 8 are not necessary for our project to work, but would be interesting extensions if we have time.

We will use the Viola-Jones face detector in OpenCV as a starting point. Once we have found two faces we will apply some post processing to make sure that we have the entire face without any holes. Once we have a binary mask of the faces’ fully connected components, we plan to use cvBlobsLib (similar to regionprops function in Matlab) to do face region labelling and extraction. Choosing the faces to swap will either be random or done through a simple UI. Initially we will translate the faces to match the new face’s centroid with the centroid of the old face. Then, using the relative locations of features detected using OpenCV (such as the eyes, nose, and mouth) we will determine the orientation of the face in the plane parallel to the camera lens. Then, the relative sizes of these features will allow us to determine the degree of rotation away from the camera. However, if the faces we are swapping have vastly different degrees of rotation toward and away from the camera it will be difficult to interpolate the faces back into the correction orientations.

When stitching the swapped faces to their new bodies, we plan to test out both Laplacian image pyramids as well as Poisson Seamless Cloning as techniques for interpolating and smoothing onto the background after moving faces around to different parts of the image. Discrete Poisson Solvers will be used for localized lighting adjustments on the transposed faces in order to match the localized background.

Main challenges we foresee are cleanly identifying the faces, finding the most natural smoothing and interpolation algorithm to stitch in the swapped faces, and the 3D face rotation and interpolation if we have additional time.
References:

IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 15, NO. 4, APRIL 2006
Seamless Image Stitching by Minimizing False Edges
Assaf Zomet, Anat Levin, Shmuel Peleg, and Yair Weiss
http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1608144

IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 10, NO. 8, AUGUST 2001
Filling-In by Joint Interpolation of Vector Fields and Gray Levels
Coloma Ballester, M. Bertalmio, V. Caselles, Associate Member, IEEE, Guillermo Sapiro,
Member, IEEE, and Joan Verder
http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=935036

Poisson Image Editing
Patrick P´erez, Michel Gangnet, Andrew Blake
Microsoft Research UK
http://portal.acm.org/citation.cfm?id=882269

CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001
Rapid Object Detection using a Boosted Cascade of Simple Features
Paul Viola, Michael Jones