Programming Assignment P1: IMAGE MOSAICING

handout 1 of 2

Rendering, 15-862

due 3 Oct. 95 (by end of day)

In this assignment you will take two or more photographs and create an image mosaic by registering, projective warping, resampling, and compositing them.

1. Shoot two or more photographs so that the transforms between them are projective (a.k.a. perspective). One way to do this is to shoot from the same point of view but with different view directions, and with overlapping fields of view. Another way to do this is to shoot pictures of a planar surface (e.g. a wall) from different points of view. I'm not particular about how you take your pictures or get them into the computer, but I recommend:

- use print film (not slides)
- shot with the same camera, lens, print film, and processing. Fisheye lenses or lenses with significant barrel distortion make the registration & warping more complicated. Any lens is ok, but wide angle lenses make more interesting mosaics.
- shot as close together in time as possible
- identical aperture & exposure time, if possible. (if your camera doesn't permit you to set or freeze the aperture AND exposure time, then shoot subjects of roughly homogeneous brightness so camera is more likely to shoot them with identical exposures)
- fields of view overlap. 20% to 50% overlap is recommended
- contain distinctive feature points or "texture" to help with registration (creating an image mosaic of a clear blue sky is boring)

If you're shooting a non-planar scene, then shoot pictures from the same position (turn camera, but don't translate it). Shoot from a tripod, if possible (this is more important if objects are close).

Good scenes are: building interiors with lots of detail, inside a canyon or forest, tall waterfalls, panoramas. The mosaic can extend horizontally, vertically, or can tile a sphere. You might want to shoot several such image sets and choose the best.

If you're shooting with a standard non-digital camera, I recommend print film (not slides) since the public-access scanner is not a slide scanner.

You're welcome to use a different type of camera, such as a digital one (e.g. Apple Quicktake) or a video camera with frame grabber, if you have access to one. (Note that the effective resolution of the Quicktake seems to be significantly less than advertized; I found that I had to filter down to 320x240 size or less to get rid of its color aliasing). If you use a lower resolution camera, then please shoot proportionally more pictures so that the final mosaic will be roughly 800-1200 pixels wide.

If you don't have a camera with which you can shoot such pictures, ask a friend or classmate for help. Talk to people in class or post to the class news group, cmu.cs.class.cs862, for help.

Shoot & process your pictures early - leave time to re-shoot in case they don't come out! Lay out your photos on a table to see approximately what the mosaic will look like. Check that you've got good photos before bothering to digitize them.

2. Digitize your photos in Wean 3509. Door is locked, but for CS grad students, your NC 58 key will get you in. Put photo in Silverscan flatbed scanner; on Mac, run Photoshop; on menu: Acquire, Silverscan; scan at 100-200 pixels per inch (recommended), if 4"x6" prints, if more, the files will be huge; crop if desired; write picture to file (TIFF format is good). It is important that you transfer all of your picture files to another machine (e.g. by Appleshare) and clean up after yourself, since these are public machines. To help out those without keys or any scanning experience, we'll schedule a day when one of you can act as "scanning supervisor" and help others do their scanning.

3. Do approximate manual registration of photos (find four or more corresponding points in each pair of overlapping images, get their coordinates (e.g. using Photoshop or UNIX xv). The simplest way to do this is to find the projective transformation between images (use least squares if you have more than four point pairs). The math for this will be covered in lecture. You can use other methods if you like, but keep in mind that the transform from one image to another is not affine, but projective.

4. Warp photos so they're registered and create an image mosaic. Instead of having one picture overwrite the other, which would lead to strong mosaic artifacts, using weighted averaging (weight zero along edge of each image, peak weight at image center). You can leave one image unwarped and warp the other image(s) into its projection, or you can warp all images into a new projection; it's up to you. Don't use point sampling during resampling; use bilinear interpolation or something better.

Try to get the registration and compositing working well enough that the seams become invisible. If the exposures or processing of your pictures differ markedly, you may find it helpful to do some color correction. Vignetting (darkening on edges of image due to camera) can also be a slight problem.

If your mosaic spans more than 180 degrees, you'll need to break your mosaic into pieces (as in cubical environment maps), or else use non-projective mappings, e.g. some kind of spherical projection or fisheye projection.

5. Turn in your original images and your final, best mosaic image in (SGI) RGB, PPM, TIFF, or JPEG formats in your directory for this assignment, which will be /afs/cs/project/classes-ph/862/student/*yourname*/p1. Mark correspondence points in the original images somehow (e.g. bright red pixels). Also turn in a README file and all your source code. Write a README of a few paragraphs explaining what you did, what the picture and source files are, and evaluate how successful you think it turned out. Give me the 3x3 projective matrix that you used to transform your picture(s).

If you would like to work in a team of two, speak to me.