Proposal

Panoramic mosaics, or image stitching, is one of the oldest problem in computer vision. The purpose of it is to merge images or photos which describe the same scenery into a single graph. Nowadays the problem has been solved quite well and widely used on digital cameras and smart phones. For example, Iphone has merged this technique into there camera application to take a panoramic scenery.

This project will base on the paper [1] and [2].

[1] gives a set of steps to merge the images:

1) Extract and match SIFT features between all of the images. To make this features scale invariance, locate SIFT features at scale-space maxima/minima of a difference of gaussian function.
2) Find all overlapping/matching images. Since the user may input bunch of images with belongs to several different panoramas, this step is to divide the images sets so that each set of images belongs to the same panoramas. Using RANSAC to get the matching inliers and use a probabilistic model to verify the match. This step will reject noise images which match to no other images.
3) Bundle adjustment. If we stitching images one by one, there will be some accumulate errors which will cost problems such as “Gap closing”. So Bundle adjustment matching the set of images which belong to single panoramas in the same time. Before doing Bundle adjustments, all images need to be initialized with same rotation and focus length as the image to which it best match. Then Bundle adjustment will minimize the squared error which is the sum of distance of match errors(squared) between every two images which have overlap part(s). [1] use Levenberg-Marquardt algorithm to find the solution of this “long” equation.
4) Multi-band Blending. After above steps, the images are roughly merged. But the edges between mosaics are obvious because of the difference of intensity in each mosaics images. Also, there are “ghost” which caused by mis-alignments(parallax). [1] and [2] use Laplacian Pyramid of the images. The algorithm do smoothing to the levels of the Laplacian pyramid (low frequency) of images. The lower the level is(higher resolution), the lower the smoothing weight is. So, by blending the low frequencies over larger spatial range and high frequencies over a short range, the images will be blended nicely. Usually only two levels are enough to compensate for difference in exposure.

[2] is an improved version of [1] which gives more descriptions of original steps(1-4) and gives some additional steps such as “Automatic Panorama Straightening”(heuristics; It assume that the images are in the similar horizontal, so the algorithm try to “straighten” the result panoramas) and “Gain Compensation”(deal with a photometric parameter, namely the overall gain between images)

One of the things that [1] and [2] don't deal with is “radial distortion”. [3] introduces the solution: Find the average weighted 3D point of the matched 2D points; Find the distance between a 2D point P in an image and the 2D point P' corresponding to the average weighted 3D point (which corresponding to the P), which is Dis(P', P); For an image, use several 2D points in it to get several

Dis(P', P) values, and use these distance values to interpolate to a matrix(same size of the image); For each point in the image, move it according to the interpolation matrix; Run these steps to all images which mis-alignments to each other; Now get the result without radial distortion.

Other problem [1] and [2] don't deal with is "Scene Motion". According to [3], this can be done by finding a consistent seam which can separate the overlapping fragment or using RODs or another algorithm by Agarwala, Dontcheva, Agrawala.

What the project will do are the four steps described in [1]. But it seems that there is much workload to implement all of it. So I plan to select some parts to implement and try to find libs or use OpenCV function to do other parts, depending on the time I spend on coding. If the time is sufficient, I will implement the other two steps described on [2].

About data: There are many panoramic dataset on web. What I find now is [4], and I plan to use it. For each panoramas it contains 4-21(what I find now) mosaics with about 500*500 image size.

[1] M. Brown and D.G. Lowe, "Recognising Panoramas" *Proc. ICCV* 2003.

[2] M. Brown and D.G. Lowe, "Automatic Panoramic Image Stitching using Invariant Features", *International Journal of Computer Vision, 2007*

[3] R. Szeliskim, "Computer Vision: Algorithms and Applications", *2011*

[4] http://www.iiia.csic.es/~aramisa/datasets/iiiapanos.html