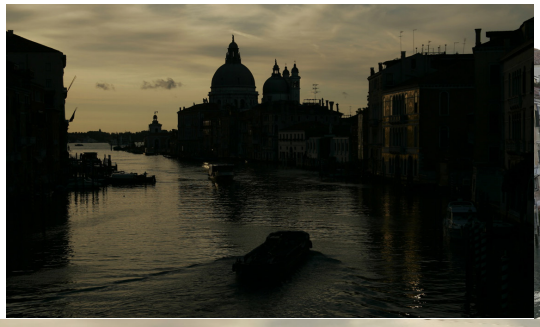
HDR Analysis

Sample Results

The results below are done with number of levels = 9.



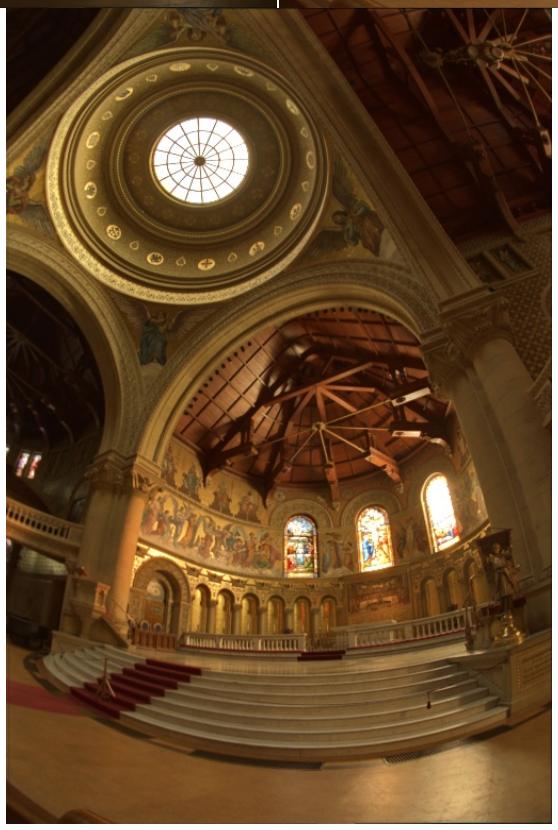
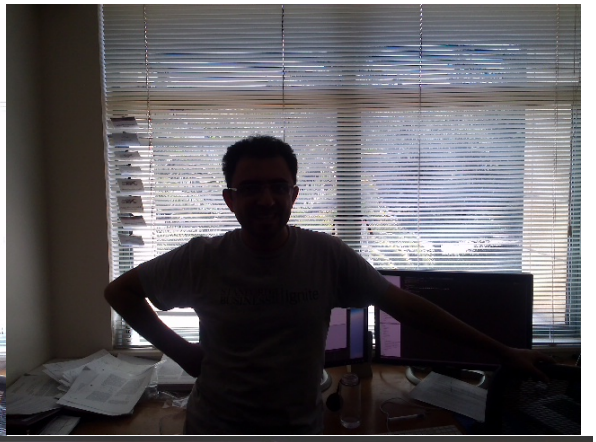


Image results from the device



If your input images had a large EV delta, how would it affect the output?

The parts of the two images with better explore will gain more weight and therefore the image with the better exposure will have a higher contribution to the final pixel value.

Compare Laplacian pyramid blending to linear blending.

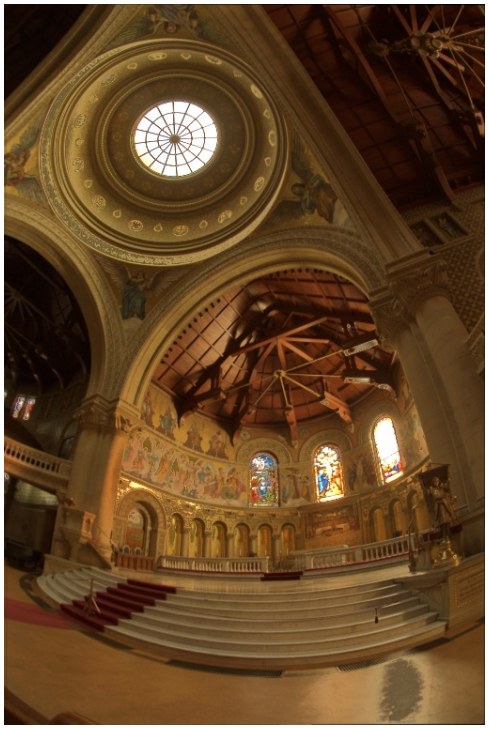
Linear blending does not enable smoothing of edges while the Laplacian pyramid blending model enables it through applying weights to lower resolution and blurred

versions of the original image. The hierarchy of images allows filtering weights and applying the key weights to the blurred version. The outcome is a less noisy and more smooth image compared to the linear blending model.

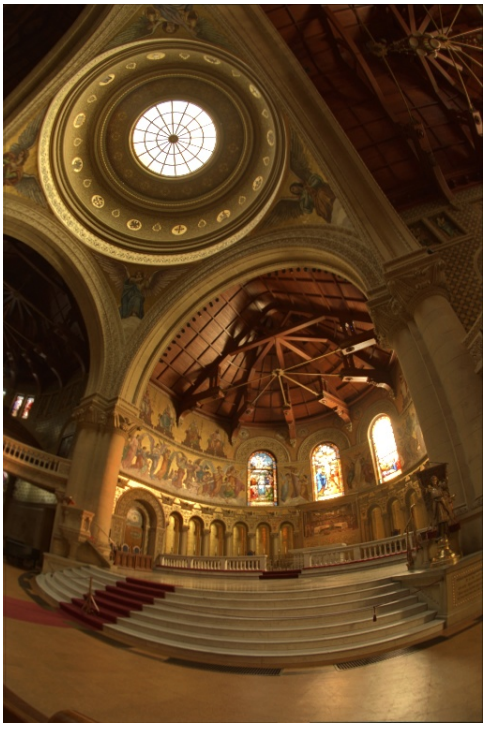
How does the number of levels in pyramid blending affect the output?

The larger the number of levels, the more smoothing appears in the image. However, beyond a certain level, depending on the image resolution, the extra levels will not lead to any significant changes to the image. This is because the image size of the last level becomes so small that not so much more smoothing can be extracted form making the image any smaller. For the difference see the two images below.

Number of levels = 1 (similar to linear blending)

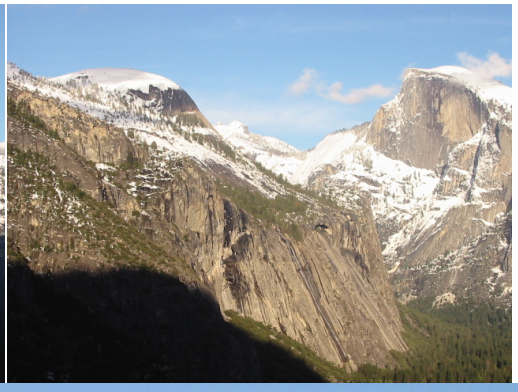


Number of levels = 9



Panorama Analysis

Sample Results



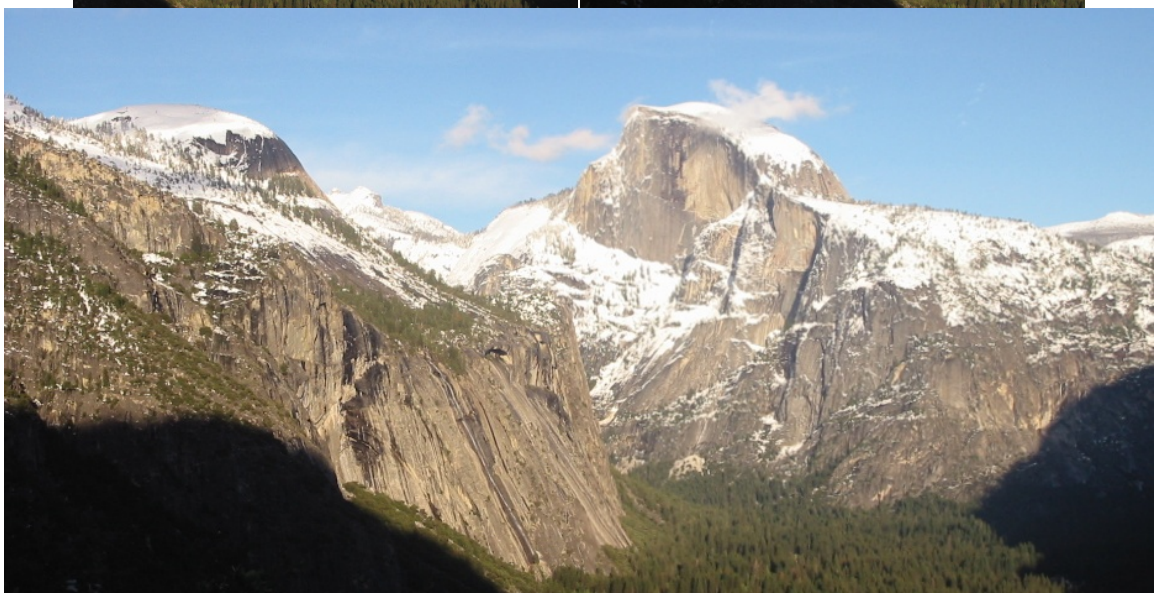


Image results from the device

In the case below, the exposure of the two consecutive images has some variation and as a result, we find that the boundary of the two images is more clearly shown. A better implementation of the Laplacian blending that takes into account saturation, exposure, and contrast eliminates this boundary.

In your live examples, how does the distance from the camera to the objects in the scene affect performance? What assumptions have we made by using a homographic transformation to map pixels? Why might this be wrong in some cases?

I observe that the closer the object, the more time it takes to process. This is because closer objects lead to more matching key-points; this in turn creates more fitting candidates for the RANSAC algorithm to try in order to find the best fit.

In this approach, we assume there exists a transformation line that can best fit the key points. So, in images that have their key points sparsely distributed all across the space (not well lined up near each other), RANSAC may end up choosing a sub-optimal transformation.

What might be some issues with multiple images for panoramas?

1) The order of images must be properly handled. The left and right image compared to each image is important.
2) We will observe cumulative image orientation error in the homography building up when going through multiple images.
3) A point can become a match for the image on the left and on the right if the angle of rotation is too small. This may cause the seam lines to cross (which is not a problem, but needs extra checking in the code to make sure it functions as expected)

How would sensor data help you with the panorama? Would acceleration and rotation information be helpful? How?

Rotation: it can help better gauge the image orientation when applying to homography function. Right now, the homography function works based on the pruned match points. The H function can inputs be replaced (or improved) using rotation information.

Acceleration: when image frames are automatically captured by the camera (e.g. the panorama feature in the iPhone), acceleration information helps capture frames at an optimal rate to maximize non-overlapping space of each image. This allows for processing fewer images to build a panoramic view of a scene.