High Dynamic Range and noise

The goal here is to combine samples with different exposure times to obtain better SNR. In our case we consider 2 samples, i.e. 2 images perfectly co-registered with 2 different exposure times that we may call LE (low exposure) and HE (high exposure).
Regarding noise, we only have photon noise.

In this case, HDR process is a weighted combination that can be depicted as follows:
1) In order to equiparates exposures, both images are converted to 16 bit irradiance.
2) we correct HE so that it can be combined with LE: corrected_HE = HE / ratio of exposures
Then the combination:
3) output pixels where HE is saturated = LE pixels
4) output pixels elsewhere = w1 * LE + w2 * corrected_HE ; w1 + w2 = 1

What would be the output image SNR in function of the LE SNR, w1, w2, and ratio of exposures?
What is the best weights formulation to optimize output SNR?

Motion Blur estimation and correction

Our satellites are orbiting around the earth with a velocity of 7km/s, our ground sample distance in our images is 1m per pixel. We may expose several hundreds of ms.
We have internal procedures to compensate its motion and avoid blur in our images. Yet, it may happen that this procedure does not achieve a 100% compensation, leading to a bit of motion blur in our images.
If we ask in google maps for the position (40.851646, 109.628882) we will be in China, at a "calibration target". Those sites are suitable to estimate motion blur for example.

Describe how you would estimate and correct it for this precise place.
What metric would you use to validate your correction ?

PSF deconvolution techniques

You may find an example of PSF (Point Spread Function) attached to this document.
In python, in a git repo, with a high quality image showcase of your choice:
1) Apply our example PSF to your image
2) Add gaussian and photon noise
3) compute a psf deconvolution of your choice.

Satellite Camera Simulator

We want to have a simulator to test geometrical algorithms. We will Simulate a satellite acquisition of a place on Earth.
We consider a spherical earth, and a pinhole camera model to represent our telescope.
Satellite has an altitude of 450km (above earth's surface), points to earth center (always at nadir).
Focal length of this pinhole camera is 2.0m, and pixel size is 5x5 micrometer.

In python, in a git repo.
1) Implement this simulator, with a method that takes, as input, a random place on earth, with ECEF coordinates, and output satellite position and pointing 3x1 vectors.
2) Add a method which considers some random points spread around an input location within a radius of 3km maximum. Project them to the 2D focal plane of the pinhole camera, it would answer an image.

Focal length Estimation

In order to Orthorectify a raw image, we consider a pinhole camera model to represent our telescope. In a first time, we assume GPS and attitude from telemetry to be perfect. We yet have to estimate the focal length of this pinhole camera.
Satellite has an altitude of 450km (above earth's surface), points to earth center (assume at nadir), and the ground sampled distance (GSD) appears to be 1m per pixel. A pixel is a 5x5 micrometer square.

1) Expressed the focal length of this pinhole camera.
2) Quantify the impact of 100m error in GPS along the axis “target - satellite” on focal length estimation.

We assume focal length is constant over time, so we can consider several images of a unique place from the same satellite at different times.
For each capture, telemetry has some random gaussian noise to GPS position.
3) Propose an algorithm to estimate the focal length.
4) If you chose to do the previous exercise, re-use the simulator you implemented to test it.