Subject: Need help reconstructing flat-field. Minimization problem.
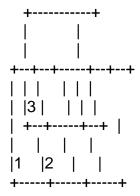Posted by JJ on Tue, 23 Oct 2007 22:16:40 GMT
View Forum Message <> Reply to Message

HELP!

I'm trying to reconstruct the flat-field information for a CCD from
overlapping images of the same scene - with same exposure duration and
same lighting conditions, but I'm failing miserably.

I'm pretty sure I've got all the information I need. The CCD is
1024x1024, and I have 3 images that in are close to the following
pattern (labeled 1,2,3 in lower left corners):

(need fixed-width font like courier)

```
     +-----------+
     |           |
     |           |
   +--+--+-----+--+--+
   | | |   | | |
   | |3|   | | |
   | +--+-----+--+ |
   | | | | |
   |1   |2   |   |
   +-----+-----+-----+
```

The flat-field is horrendous - with a variation of more than a factor of
two from brightest (middle left of image) to darkest (right edge).

(For those not familiar with flat-fields, the flat field can be
considered as the response function of the CCD/camera system. If you
image a perfectly homogeneous source, the image you would get is the
flat field - plus noise. If you know the flat-field, you can divide it
out of the image to give you the correct brightness of each pixel in the
scene.)

So, for each place where two images above overlap, we know that the
scene brigthness should be the same (within the noise) and we can
calculate the ratio of the actual difference in the images.

For example: img1[x1,y1] / img2[x2,y2] = R[x1,y1,x2,y2]

Since the scene brightness for that pixel should be the same in both
images, we can say that

img1[x1,y1]   img2[x2,y2]
----------- = -----------

flat[x1,y1]   flat[x2,y2]

So, we know that

flat[x1,y1]
----------- = R[x1,y1,x2,y2]
flat[x2,y2]

For these 3 overlapping images, I get ~ 1.4 million equations, each
giving the ratio between 2 pixels in the flat field.

Great!

But I can't seem to solve for the best flat-field image that will
minimize the residuals for this set of equations. This is clearly too
large a set of parameters to pass to a minimization routine such as
POWELL or mpfit, and the flat-field is so splotchy and irregular that I
don't know how to model it with a 2-D function such that I could
minimize the number of parameters and actually solve it with a
minimization routine.

I did write a routine that will step through the 1.4 million equations,
and appropriately adjust the flat field for cases where the ratio is not
already determined by the previous equations. For example if A/B = R1,
B/C = R2, C/D = R3 (where A = flat[Ax,Yx], etc.) if we then get D/A =
R4, that is a conflict, because that value of D/A is already determined
by the previous equations. My function ignored about 400k equations,
and the result, was a pretty much complete image that didn't look at all
like the flat field but seemed to have a lot of structure that was
related to the way the images overlapped.

I think the problem with this method, is that it propagates and
multiplies errors due to noise - which is why I think I need to minimize
the residuals of all equations together rather than just throwing out
some of the equations. Apart from the fact that it returned junk, the
algorithm seemed OK. When I tried it with faked data with no noise, it
worked perfectly. When I tried it with the same faked data with noise,
I get the same junk.

Any help on how to tackle this problem greatly appreciated.

Thanks.

-Jonathan Joseph