Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by Tal on Wed, 24 Oct 2007 08:46:41 GMT
View Forum Message <> Reply to Message

>> Apart from the fact that it returned junk, the algorithm seemed OK.

sounds familiar.
:-)

Hi Jonathan,

If i got all this right then there is one thing missing here: Dark current images. Take image for the same location of your light images 1,2,3 but with the shutter closed. You will record in fact the noise of your camera. an integration of heat, electronics, inter-CCD variabilities in response etc.

now all you need to do is generally: (light_image - dark_image) and continue from there as usual. since your CCD camera is a 2D CCD array, following your notation the ratio is now:

img'=(img-dark)
img'1[x1,y1] / img'2[x2,y2] = R'[x1,y1,x2,y2]

where R' is the dark-corrected Ratio image R.

to be on the safe side, repeat your light measurements and dark measurements again if possible, since dark-current images are variable in time according to conditions during acquisition.

hope this helps,
Tal

Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by JJ on Wed, 24 Oct 2007 14:51:41 GMT
View Forum Message <> Reply to Message

Thanks,

But unfortunately that doesn't get me any closer to the solution. I basically already have the dark-corrected image. Removing the dark current (which I am more familiar with that I want to be - believe me) does not remove the shot noise, which scales with the square root of the # photons hitting the pixel. This noise is I believe the result of the quantum nature of light. So imaging the same exact point twice under the same exact conditions even with a perfect camera will give you different results.

Just for argument's sake, let's assume that the dark current is 0 and
our camera is perfectly noiseless. Our CCD pixel increments by one
count for every 50 photons that hit it. For a pixel value around 2000,
you still get a shot noise of +/- sqrt(2000 * 50) / 50, which gives
roughly 2000 +/- 6 (I think that's 1 sigma).

You can't get rid of or model the shot noise because it's truly random.
And because of that, I believe that my problem requires a
minimization solution - or at least a solution that does not multiply
this noise into huge errors as it progresses, as my solution seems to.

Thanks.

-Jonathan

Tal wrote:

\>
\> sounds familiar.
\> :-)
\>
\> Hi Jonathan,
\>
\>   If i got all this right then there is one thing missing here: Dark
\> current images. Take image for the same location of your light images
\> 1,2,3 but with the shutter closed. You will record in fact the noise
\> of your camera. an integration of heat, electronics, inter-CCD
\> variabilities in response etc.
\>
\> now all you need to do is generally: (light_image - dark_image)
\> and continue from there as usual. since your CCD camera is a 2D CCD
\> array, following your notation the ratio is now:
\>
\> img'=(img-dark)
\> img'1[x1,y1] / img'2[x2,y2] = R'[x1,y1,x2,y2]
\>
\> where R' is the dark-corrected Ratio image R.
\>
\> to be on the safe side, repeat your light measurements and dark
\> measurements again if possible, since dark-current images are variable
\> in time according to conditions during acquisition.
\>
\> hope this helps,
\> Tal
\>
\>

Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by b_gom on Wed, 24 Oct 2007 18:26:42 GMT
View Forum Message <> Reply to Message

Maybe I'm not following exactly, but it seems that you are trying to correct for variations in responsivity across your CCD, and your problem is that you are trying to do this correction using data that is too noisy. Assuming the array responsivity is fixed (or at least slowly varying), can you not just build up a high S/N data set by staring at a known flat field source, and then use this data set to produce your calibration (after dark correction), or at least fit a 2D surface and use that? In other words, can't you make 1000 images instead of 3, and have them all overlap exactly?

On Oct 23, 4:16 pm, Jonathan Joseph <j...@cornell.edu> wrote:

```
> 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

Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by JJ on Wed, 24 Oct 2007 20:44:24 GMT
View Forum Message <> Reply to Message

b_gom@hotmail.com wrote:
> Maybe I'm not following exactly, but it seems that you are trying to
> correct for variations in responsivity across your CCD, and your
> problem is that you are trying to do this correction using data that
> is too noisy. Assuming the array responsivity is fixed (or at least
> slowly varying), can you not just build up a high S/N data set by
> staring at a known flat field source, and then use this data set to
> produce your calibration (after dark correction), or at least fit a 2D
> surface and use that? In other words, can't you make 1000 images
> instead of 3, and have them all overlap exactly?

b_gom@hotmail.com wrote:

> Maybe I'm not following exactly, but it seems that you are trying to
> correct for variations in responsivity across your CCD, and your
> problem is that you are trying to do this correction using data that
> is too noisy. Assuming the array responsivity is fixed (or at least
> slowly varying), can you not just build up a high S/N data set by
> staring at a known flat field source, and then use this data set to
> produce your calibration (after dark correction), or at least fit a 2D
> surface and use that? In other words, can't you make 1000 images
> instead of 3, and have them all overlap exactly?
>

OK, I'll spill the beans.

Sure, your suggestion would be nice but it's not currently feasible.
The camera's already on Mars you see:-). There's been contamination
that has caused the flat field to change drastically since the initial
calibration on Earth. And there's nothing particularly flat to look at.
In the past, we have used the sky, looking away from the sun for

flat-field monitoring and updates. On clear days, that portion of the sky could be fit as a linear gradient and the gradient could then be removed. But we're currently in the middle of a prolonged dust storm and even the sky's not so flat, and is quite variable. Even if the sky did have a nice smooth gradient, we don't have a good model for what it is based on wavelength and phase angle. Previously, a gradient was simply fit to the image and removed, the flat-field not being so bad and having no gradient of its own, this worked pretty well. But now, the flat field itself clearly has a strong gradient component to it, making it difficult to deconvolve the gradient due to the sky and the gradient due to the flat field.

Ther are other problems too - namely light scattering issues, which are complicating the problem, but it would require more data-volume that we have availalbe right now to try and model that.

For now, I'm left to work with what we have to try and come up with a reasonable flat field with which to better calbirate the images. So I'm pursuing some non-traditional approaches to this problem.

Scant though it is, I think I have the data neccessary to produce a reasonable flat field - I just can't seem to extract that flat field from the data. So now that you know the extent of the problem, I'm hoping for some suggestions on how to work with the data I have.

Right now, vaving images that don't overlap exactly seems to be giving me more information that having perfectly overlapping images. I really think there's enough information to derive a good approximation of the flat field.

So if anyone can help me solve this problem, I promise honorable mention to the camera team.

JD, are you still out there?

I'm also open to suggestions for how best to model the flat field as a surface, using few enough variables that I can reasonably use mpfit or some other minimizer to solve for them.

Thanks again.

-Jonathan

Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by JJ on Wed, 24 Oct 2007 20:45:14 GMT
View Forum Message <> Reply to Message

b_gom@hotmail.com wrote:

> Maybe I'm not following exactly, but it seems that you are trying to
> correct for variations in responsivity across your CCD, and your
> problem is that you are trying to do this correction using data that
> is too noisy. Assuming the array responsivity is fixed (or at least
> slowly varying), can you not just build up a high S/N data set by
> staring at a known flat field source, and then use this data set to
> produce your calibration (after dark correction), or at least fit a 2D
> surface and use that? In other words, can't you make 1000 images
> instead of 3, and have them all overlap exactly?
>

OK, I'll spill the beans.

Sure, your suggestion would be nice but it's not currently feasible. The camera's already on Mars you see:-). There's been contamination that has caused the flat field to change drastically since the initial calibration on Earth. And there's nothing particularly flat to look at. In the past, we have used the sky, looking away from the sun for flat-field monitoring and updates. On clear days, that portion of the sky could be fit as a linear gradient and the gradient could then be removed. But we're currently in the middle of a prolonged dust storm and even the sky's not so flat, and is quite variable. Even if the sky did have a nice smooth gradient, we don't have a good model for what it is based on wavelength and phase angle. Previously, a gradient was simply fit to the image and removed, the flat-field not being so bad and having no gradient of its own, this worked pretty well. But now, the flat field itself clearly has a strong gradient component to it, making it difficult to deconvolve the gradient due to the sky and the gradient due to the flat field.

Ther are other problems too - namely light scattering issues, which are complicating the problem, but it would require more data-volume that we have availalbe right now to try and model that.

For now, I'm left to work with what we have to try and come up with a reasonable flat field with which to better calibrate the images. So I'm pursuing some non-traditional approaches to this problem.

Scant though it is, I think I have the data neccessary to produce a reasonable flat field - I just can't seem to extract that flat field from the data. So now that you know the extent of the problem, I'm hoping for some suggestions on how to work with the data I have.

Right now, vaving images that don't overlap exactly seems to be giving me more information that having perfectly overlapping images. I really think there's enough information to derive a good approximation of the flat field.

So if anyone can help me solve this problem, I promise honorable mention to the camera team.

JD, are you still out there?

I'm also open to suggestions for how best to model the flat field as a surface, using few enough variables that I can reasonably use mpfit or some other minimizer to solve for them.

Thanks again.

-Jonathan

Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by jdshaw on Wed, 24 Oct 2007 21:06:24 GMT
View Forum Message <> Reply to Message

Jonathan,

I had a similar problem with a sky survey I did last year. The flats the telescope operators provided were well below the linear response area of the chip. So, I don't know if this will help, but this is what I did:

> From each night's data, I took about forty different images used the SKY routine from the asto libraries (http://idlastro.gsfc.nasa.gov/contents.html) to find the sky background and then normalized each image to its average sky value. I placed these in a 2048x2048x40 array (2048x2048 is the size of the chip). Then took the median value for each pixel (alternatively, you can sigma-clip out high values and average the rest).

This is simple but worked surprisingly well at preserving the flux values.

If you are taking terrestrial - sorry aereal - images this probably is to crude for your needs but may be a place to start. I suspect you may need to write your own 'SKY' routine to find suitable normalization values.

I'd be interested to hear your solution.

- John Shaw
Department of Physics and Astronomy
University of Delaware

Subject: Re: Need help reconstructing flat-field. Minimization problem.
Posted by pgrigis on Thu, 25 Oct 2007 08:13:42 GMT
View Forum Message <> Reply to Message

Just one quick suggestion, since it seems you have a method
which fails for some reason: try and see if your method works
when you rebin the data to a very small array (say, 64x64 or
even 16x16). Now this may well be so coarse as to be useless for
scientific data analysis, and hide all the fine scale details of
your flat field, but it should give you a feeling whether your
method is working the way you intended or not. If you can make
it work in that case, you may step up your resolution and see
what happens. Of course it is possible that these introduces
new issues which have to be dealt with, but at least you are
taking it one step at a time...

Ciao,
Paolo

Jonathan Joseph wrote:
> b_gom@hotmail.com wrote:
>> Maybe I'm not following exactly, but it seems that you are trying to
>> correct for variations in responsivity across your CCD, and your
>> problem is that you are trying to do this correction using data that
>> is too noisy. Assuming the array responsivity is fixed (or at least
>> slowly varying), can you not just build up a high S/N data set by
>> staring at a known flat field source, and then use this data set to
>> produce your calibration (after dark correction), or at least fit a 2D
>> surface and use that? In other words, can't you make 1000 images
>> instead of 3, and have them all overlap exactly?
>
>
>
>
>
> b_gom@hotmail.com wrote:
>
>> Maybe I'm not following exactly, but it seems that you are trying to
>> correct for variations in responsivity across your CCD, and your
>> problem is that you are trying to do this correction using data that
>> is too noisy. Assuming the array responsivity is fixed (or at least
>> slowly varying), can you not just build up a high S/N data set by
>> staring at a known flat field source, and then use this data set to
>> produce your calibration (after dark correction), or at least fit a 2D
>> surface and use that? In other words, can't you make 1000 images
>> instead of 3, and have them all overlap exactly?
>>
>
> OK, I'll spill the beans.

>
> Sure, your suggestion would be nice but it's not currently feasible.
> The camera's already on Mars you see:-). There's been contamination
> that has caused the flat field to change drastically since the initial
> calibration on Earth. And there's nothing particularly flat to look at.
> In the past, we have used the sky, looking away from the sun for
> flat-field monitoring and updates. On clear days, that portion of the
> sky could be fit as a linear gradient and the gradient could then be
> removed. But we're currently in the middle of a prolonged dust storm
> and even the sky's not so flat, and is quite variable. Even if the sky
> did have a nice smooth gradient, we don't have a good model for what it
> is based on wavelength and phase angle. Previously, a gradient was
> simply fit to the image and removed, the flat-field not being so bad and
> having no gradient of its own, this worked pretty well. But now, the
> flat field itself clearly has a strong gradient component to it, making
> it difficult to deconvolve the gradient due to the sky and the gradient
> due to the flat field.
>
> Ther are other problems too - namely light scattering issues, which are
> complicating the problem, but it would require more data-volume that we
> have availalbe right now to try and model that.
>
> For now, I'm left to work with what we have to try and come up with a
> reasonable flat field with which to better calbirate the images. So I'm
> pursuing some non-traditional approaches to this problem.
>
> Scant though it is, I think I have the data neccessary to produce a
> reasonable flat field - I just can't seem to extract that flat field
> from the data. So now that you know the extent of the problem, I'm
> hoping for some suggestions on how to work with the data I have.
>
> Right now, vaving images that don't overlap exactly seems to be giving
> me more information that having perfectly overlapping images. I really
> think there's enough information to derive a good approximation of the
> flat field.
>
> So if anyone can help me solve this problem, I promise honorable mention
> to the camera team.
>
> JD, are you still out there?
>
> I'm also open to suggestions for how best to model the flat field as a
> surface, using few enough variables that I can reasonably use mpfit or
> some other minimizer to solve for them.
>
> Thanks again.
>
> -Jonathan