Subject: Re: Reconstruct surface from gradient field?
Posted by dg86 on Sat, 15 Feb 2014 02:42:53 GMT
View Forum Message <> Reply to Message

On Friday, February 14, 2014 8:26:25 PM UTC-5, Craig Markwardt wrote:
> On Friday, February 14, 2014 8:22:46 PM UTC-5, Craig Markwardt wrote:
>
>> On Thursday, February 13, 2014 5:24:22 PM UTC-5, David Grier wrote:
>
>
>
>> There might be some heuristic way to iterate towards the solution, I'm not sure. Poisson's equation has a nice solution like that, but you don't have a Poisson's equation unless you take a (even more noisy) derivative of your measured data.
>
>
>
> This page might have some interesting ideas.
>
> http://dsp.stackexchange.com/questions/2859/how-do-i-numeric ally-calculate-a-function-from-its-noisy-gradient
>
>
>
> ... especially the Frankot-Chellappa discussion. If you solve this by FFT, you will still need some kind of spatial filter to de-weight the measurement noise.
>
>
>
> Craig

Dear Craig,

Thanks for all of your helpful suggestions. Your comments about noise and the Poisson problem were spot on, and led to a better solution than I'd found previously. Rather than computing the divergence of the gradient field by finite differences, I'm calculating it in reciprocal space, with noise suppression:

$$Z(kx,ky) = -i\,[kx\,Gx(kx,ky) + ky\,Gy(kx,ky)] / [k^2 + eps^2]$$

Z(kx,ky) is the Fourier transform of the desired solution. Gx and Gy are the Fourier transforms of the gradients, computed with FFT(). The key thing is to choose the parameter eps to suppress noise. It's sort of like a Wiener filter. The solution then is the real part of the inverse FFT of Z(kx,ky).

This is a lot better than my previous effort, but still not good enough. Next, I'm going to try your least-squares approach.

TTFN,

David