
Subject: Re: Reconstruct surface from gradient field?
Posted by [Craig Markwardt](#) on Sat, 15 Feb 2014 01:22:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thursday, February 13, 2014 5:24:22 PM UTC-5, David Grier wrote:
> Dear Folks,
>
>
>
> I have measurements of the gradient field (tangent vectors) associated with
>
> a two-dimensional surface, and would like to compute the height of the surface
>
> itself. I understand that there are standard algorithms to do this, and wonder if
>
> any of them have been implemented in IDL. If so, I'd be grateful for a pointer.

I don't have any existing code.

I think you should look at this as a least squares problem. You have measurements, presumably with some measurement error, and you want a surface that fits those measurements within the error. I think that is best expressed as a least squares (fitting) problem.

In least squares, you are trying to minimize the chi square value,
$$\text{CHI_SQUARE} = \text{TOTAL}((\text{grad_x}(x_j, y_j) - \text{grad_x_measured}(x_j, y_j))^2) + \$$$
$$\text{TOTAL}((\text{grad_y}(x_j, y_j) - \text{grad_y_measured}(x_j, y_j))^2)$$
where XJ and YJ are the (x,y) position of every measured point and GRAD_{X,Y} is the model-calculated gradient and GRAD_{X,Y}_MEASURED is the measured value. You can easily add measurement errors to this equation in the standard way for chi-square.

Now it's "just" a matter of defining a function F whose gradients GRAD_X and GRAD_Y you can calculate, and then adjusting the the function until you get a good fit by minimizing CHI_SQUARE. Sounds easy!

You will need to consider the nature of your surface. There are an infinite number of surfaces F that will produce your measured gradients, to within the measurement range. Do you need some smoothness to your function? Is a bilinear interpolation enough? Does it need to have continuous derivatives everywhere? Do you need the function on a regular grid or an irregular grid?

Let's say you have a small regular grid of NX x NY points and you are satisfied with a piecewise bilinear function (small flat plate segments between the points). Then this is really a function with NX x NY parameters, and you want to solve for each of these parameters. NX x NY better be small or you will run into problems.

Then the gradient at any point (xj,yj) will be a simple finite difference,
$$\text{grad_x}(x_j, y_j) = (\text{F}(x_j + dx, y_j) - \text{F}(x_j - dx, y_j)) / (2 * dx)$$

and similar for GRAD_Y. If you substitute this back into the expression for CHI_SQUARE, you have an equation you want to minimize that depends on the measured values and the parameterized F() function.

To solve this as a least squares problem, you will take the derivative of CHI_SQUARE with respect to each parameter and set to zero, which is the standard way to solve least squares problems (see Numerical Recipes). That gives you $N_X \times N_Y$ equations. For this simple bilinear problem, you will get a very sparse matrix that depends only on $(i-2, i, i+2)$ in each $\{X, Y\}$ direction.

A standard least squares solver can do this as long as the number of parameters is small. In fact, if the number of parameters is small, you can choose from some nice functions like bicubic spline or thin plate splines which will give you a smooth behavior. (Even better if you can just put the control points for your spline at the places where you know the function is varying). You can easily compute the gradients by finite difference techniques.

Otherwise you will need to develop a specialized matrix solver which can handle sparseness of this kind. Since X and Y are coupled, it's sparse, but there will be lots of off-diagonal terms. (you can't put both X and Y on the diagonal at the same time)

I said it was "just" a least squares problem, but if you have a large grid, it could get problematic!

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.

Craig
