paszynsk@agh.edu.pl

Derivation of the system of linear equations

Brak plików do pobrania.

The problem of selecting the linear combination coefficients of the B-spline function used for bitmap approximation (the problem of scaling individual B-splines) is a global problem, and it should be solved taking into account all the coefficients simultaneously.

To this end, we conduct the following reasoning, which is the basis of the intuition underlying the finite element method.

Wyprowadzenie 1: Derivation of a system of linear equations for the problem of isogeometric projection

Our goal is to approximate a bitmap using a slender function u
$u(x,y) \approx BITMAP(x,y)$
Our slim feature u it is formed by the linear combination of many base functions
$u(x,y) = \sum_{i=1,j=1}^{N_x,N_y} u_{i,j} B^x_i(x) B^y_j(y)$ where $u(x,y) =_j (y)$ are the coefficients of this combination, the values of which we must calculate.

Our base functions are spread over $N_x * N_y$ mesh elements. So there are also these functions $N_x * N_y$, and therefore we need to calculate $N_x * N_y$ coefficients $u_{i,j}$. Each of these coefficients $u_{i,j}$ tells me how to scale one base function, a two dimensional Bspline function $B_{i,j;2}(x,y) = B_i^x(x)B_j^y(y)$.

How do I calculate these coefficients

$\{u_{i,j}\}i = 1, .., N_x; j = 1, \dots, N_y$.

Instead of choosing pixel values, we will use the averaging method. We also use our B-spline base functions for averaging. We want our approximation to approximate the bitmap $u(x,y) \approx BITMAP(x,y)$.

So let's choose any one B-spline function

$v(x,y) = B_{k,l;2}(x,y) = B_k^x B_l^y$. We chose a function with indices k, l. For simplicity, we named it v.

Three exemplary selections of our B-spline function are illustrated in the figure . We will call this function a test function. This is the standard nomenclature used in the finite element method.

We then use our test function to compute the average of the bitmap surrounding the function.

In other words, we want our approximation u it approximated our bitmap with such averaging distribution as our test function gives us. To write this operation mathematically, we multiply our approximation equations by the test function and calculate the integral

$\int_\Omega u(x,y)v(x,y) = \int_\Omega BITMAP(x,y)v(x,y)$.

The focal point has the greatest impact on the value of this mean, according to the appearance of the B-spline base function, and the more distant cand points have a smaller effect on the mean value.

Now note that one selection of the test function {OPENAGHMATHJAX (type = "inline" anchor = "eq2: 6" isNumeration = "yes")} v {OPENAGHMATHJAX} gives us one equation.

How many ways can we choose our test functions? Since each of them is a Bspline associated with the center of an element, and there is an element's $N_x * N_y$ it means we can choose $N_x * N_y$ test functions.

Each of these selections of test functions gives us one equation, so we get a layout $N_x * N_y$ równań

$$\int u(x,y)B_1^x(x)B_1^y(y) = \int BITMAP(x,y)B_1^x(x)B_1^y(y)$$
$$\int u(x,y)B_1^x(x)B_2^y(y) = \int BITMAP(x,y)B_1^x(x)B_2^y(y)$$
$$\vdots$$
$$\int u(x,y)B_k^x(x)B_l^y(y) = \int BITMAP(x,y)B_k^x(x)B_l^y(y)$$

$$\vdots$$

$$\int u(x,y)B^x_{N_x}(x)B^y_{N_y-1}(y) = \int BITMAP(x,y)B^x_{N_x}(x)B^y_{N_y-1}(y)$$
$$\int u(x,y)B^x_{N_x}(x)B^y_{N_y}(y) = \int BITMAP(x,y)B^x_{N_x}(x)B^y_{N_y}(y).$$

The question that needs to be asked now is that of the unknowns in our system of equations. We have $N_x * N_y$ equations, but where are the unknowns? What are the unknowns in our system of equations? To reach the unknowns, remember how our function is constructed u approximating a bitmap.

$$u(x,y) \approx \sum_{i,j} u_{i,j} B^x_i(x) B^y_j(y)$$

Our unknowns are coefficients $\{u_{i,j}\} i = 1, \ldots, N_x, j = 1, \ldots, N_y$
specifying how we "pull" individual B-splines "hills" in order to obtain the best bitmap approximation.
So we put our linear combination of B-spline in place u and we get

$$\int \sum_{i,j} u_{i,j} B^x_i(x) B^y_j(y) B^x_1(x) * B^y_1(y) = \int BITMAP(x,y) B^x_1(x) * B^y_1(y)$$

$$\int \sum_{i,j} u_{i,j} B^x_i(x) B^y_j(y) B^x_1(x) * B^y_2(y) = \int BITMAP(x,y) B^x_1(x) * B^y_2(y)$$

$$\vdots$$

$$\int \sum_{i,j} u_{i,j} B^x_i(x) B^y_j(y) B^x_k(x) * B^y_l(y) = \int BITMAP(x,y) B^x_k(x) * B^y_l(y)$$

$$\vdots$$

$$\int \sum_{i,j} u_{i,j} B^x_i(x) B^y_j(y) B^x_{N_x}(x) * B^y_{N_y-1}(y) = \int BITMAP(x,y) B^x_{N_x}(x) * B^y_{N_y-1}(y)$$

$$\int \sum_{i,j} u_{i,j} B^x_i(x) B^y_j(y) B^x_{N_x}(x) * B^y_{N_y}(y) = \int BITMAP(x,y) B^x_{N_x}(x) * B^y_{N_y}(y)$$

We now use the algebraic property of the system of equations, which allows us to extract the sum with the coefficients over the individual integrals

$$\sum_{i,j} u_{i,j} \int B^x_i(x) B^y_j(y) B^x_1(x) * B^y_1(y) = \int BITMAP(x,y) B^x_1(x) * B^y_1(y)$$

$$\sum_{i,j} u_{i,j} \int B^x_i(x) B^y_j(y) B^x_1(x) * B^y_2(y) = \int BITMAP(x,y) B^x_1(x) * B^y_2(y)$$

$$\vdots$$

$$\sum_{i,j} u_{i,j} \int B^x_i(x) B^y_j(y) B^x_k(x) * B^y_l(y) = \int BITMAP(x,y) B^x_k(x) * B^y_l(y)$$

$$\vdots$$

$$\sum_{i,j} u_{i,j} \int B^x_i(x) B^y_j(y) B^x_{N_x}(x) * B^y_{N_y-1}(y) = \int BITMAP(x,y) B^x_{N_x}(x) * B^y_{N_y-1}(y)$$

$$\sum_{i,j} u_{i,j} \int B^x_i(x) B^y_j(y) B^x_{N_x}(x) * B^y_{N_y}(y) = \int BITMAP(x,y) B^x_{N_x}(x) * B^y_{N_y}(y)$$

Note now that the above system of equations can be written in matrix form.
Especially marked in red współczynniki u_i, j form a vector of unknowns, integrals $\int B^x_i(x) B^y_j(y) B^x_k(x) * B^y_l(y) dx$ integrals form a matrix whose lines are numbered $i = 1, \ldots, N_x; j = 1, \ldots, N_y$, and columns from $k = 1, \ldots, N_x; l = 1, \ldots, N_y$. This is because the integrals contain the products of two two-dimensional B-spline functions, two "hills" spanning a finite element mesh. The first B-spline function, highlighted in black, is used for approximation, while the second B-spline function, highlighted in blue, is used to sample the bitmap (to compute the average of the bitmap). The vector of the right side is the vector of integrals in which the bitmap is multiplied by the "blue" test B-spline used for averaging, and decomposed according to the distribution described by our test B-spline.
In summary, we have thus obtained the following system of equations in which the unknowns are the coefficients $\{u_{i,j}\} i = 1, \ldots, N_x; j = 1, \ldots N_y$

$$\begin{bmatrix} \int B^x_{1,p} B^y_{1,p} B^x_{1,p} B^y_{1,p} & \int B^x_{1,p} B^y_{1,p} B^x_{2,p} B^y_{1,p} & \cdots & \int B^x_{1,p} B^y_{1,p} B^x_{N_x,p} B^y_{N_y,p} \\ \int B^x_{2,p} B^y_{1,p} B^x_{1,p} B^y_{1,p} & \int B^x_{2,p} B^y_{1,p} B^x_{2,p} B^y_{1,p} & \cdots & \int B^x_{2,p} B^y_{1,p} B^x_{N_x,p} B^y_{N_y,p} \\ \vdots & \vdots & \vdots & \vdots \\ \int B^x_{N_x,p} B^y_{N_y,p} B^x_{1,p} B^y_{1,p} & \int B^x_{N_x,p} B^y_{N_y,p} B^x_{2,p} B^y_{1,p} & \cdots & \int B^x_{N_x,p} B^y_{N_y,p} B^x_{N_x,p} B^y_{N_y,p} \end{bmatrix} \begin{bmatrix} u_{1,1} \\ u_{2,1} \\ \vdots \\ u_{N_x,N_y} \end{bmatrix}$$

$$= \begin{bmatrix} \int BITMAP(x,y) B^x_1(x) * B^y_1(y) dx \\ \int BITMAP(x,y) B^x_1(x) * B^y_2(y) dx \\ \vdots \\ \int BITMAP(x,y) B^x_{N_x}(x) * B^y_{N_y}(y) dx \end{bmatrix}$$

In a "magical" way, solving the above system of equations will give us the best approximation factors for the bitmap through our B-splines.

The aforementioned "magic" results here from the way of determining individual equations of our system, in which we postulated that on average, in the sense of the distribution determined by the B-spline testing functions, our approximation will correspond to the bitmap.

Utworzona przez admin. Ostatnia aktualizacja: Środa 07 z Październik, 2020 07:55:01 UTC przez paszynsk@agh.edu.pl. Autor: Maciej Paszynski

STATUS: W opracowaniu Zgłoś do recenzji Edytuj