Surfaces Reconstruction Via Inertial Sensors for Monitoring

Nathalie Saguin-Sprynski, Laurent Jouanet, Bernard Lacolle, Luc Biard

▶ **To cite this version:**

Nathalie Saguin-Sprynski, Laurent Jouanet, Bernard Lacolle, Luc Biard. Surfaces Reconstruction Via Inertial Sensors for Monitoring. EWSHM - 7th European Workshop on Structural Health Monitoring, IFFSTTAR, Inria, Université de Nantes, Jul 2014, Nantes, France. pp.702-709. hal-01020418

HAL Id: hal-01020418

https://inria.hal.science/hal-01020418v1

Submitted on 8 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

7th European Workshop on Structural Health Monitoring
July 8-11, 2014. La Cité, Nantes, France

SURFACES RECONSTRUCTION VIA INERTIAL SENSORS FOR MONITORING

Nathalie Saguin-Sprynski[1], Laurent Jouanet[1], Bernard Lacolle[2], Luc Biard[2]

[1] *Univ. Grenoble Alpes, F-38000 Grenoble, France*
CEA, LETI, Dépt DSIS, Minatec Campus, F-38054 Grenoble, France.
[2] *LJK, UJF , F-38041 Grenoble, France*

Nathalie.Saguin@cea.fr

ABSTRACT

This document deals with the new capabilities of monitoring via the surface reconstruction of stuctures with sensors' arrays systems. Indeed, we will detail here our new demonstrator composed of a smart textile equipped with inertial sensors and a set of processings allowing to reconstruct the shape of the textile moving along time. We show here how this new tool can provide very useful information from the structures.

KEYWORDS : *Surface reconstruction, shape capture, geometric models, inertial sensors.*

INTRODUCTION

What is targeted here is to introduce new kinds of instrumented materials. We think for example about plastic or textile surfaces, which will be equipped with arrays of sensors in order to gain some new properties. The alliance between instrumented materials and mathematical algorithms will allow materials be able to access some knowledge about their own shape, introducing what we could call proprioceptive materials. These smart materials could then become useful for monitoring structures as bridges or dams.

In this paper, we first describe the context of the shape capture and the interest for the Structural Health Monitoring. We then detail our system called MorphoShape, decomposing first the physical smart surface and its sensors, then the algorithms allowing the system to provide its own shape. We finally show virtual reconstructions of our smart surface in the graphical interface developed.

1. SHAPE RECONSTRUCTION AND SHM

CEA-Leti has developed for many years microsensors (micro-objects able to give local information), and especially inertial sensors, which are able to provide data about thier orientation by respect to some fields. A main question studied in our department is how to use a huge quantity of local information in order to have global characteristics. This led to the development of the motion capture when using some sensors in well known configurations (like following the human movements for instance), or what is interesting us in this paper, which is the shape capture of objects when we can use a lot of sensors without precisely a priori information about the shape nor the behavior.

In this context, our goal is to use inertial sensors distributed on a surface in order to retrieve the shape of this surface moving along time. Previous works answered the question of curves reconstruction (via a ribbon of sensors, named Morphosense)(e.g. [1, 2]), and the question of surface reconstruction via some ribbons laid on it (e.g. [3, 4]). We focus now on the reconstruction of an instrumented surface, thus a surface where the set of sensors is distributed in a squared mesh.

Having a smart surface brings a huge set of capabilities in the domain of SHM. Indeed, the structural health monitoring is consisting especially in detecting evolutions or modifications of structures, and following their shape can then bring a new modality for knowing. This monitoring can be declined

Copyright © Inria (2014)

in two options: either as new monitoring tools, in deploying our smart system for special test sessions; either in integrating sensors directly in the structures and then having regular data from the structure.

Thereby, the methodology for using shape capture for SHM can be as following: for a defined structure, we determine useful parameters from the shape we must follow. That will furnish the kind of instrumentation we need to realise (number of sensors and their modality, acquisition frequency, size of the system, integrated to the structure or specific external tool). Then, we can provide the shape or some specific parameters along time that may detect first signs of failure.

2. MORPHOSHAPE - PHYSICAL SURFACE

We present here our first demonstrator of surface able to provide its shape, named MorphoShape.

2.1 The sensors

The sensors used here are inertial sensors, which are sensors able to provide their orientation towards the field they measure. In our case, we use:

- microaccelerometers, which are able to provide the angle between the sensor and the vertical (Earth's Gravity field) (as long as the sensor is quite static).
- micromagnetometers, which are able to provide angular information with the Earth's magnetic field (when no magnetic pertubations around them occur).

These sensors have to be combined as a biaxial or a triaxial way. We use sensors' nodes composed of 3-axes accelerometers, and 3-axes magnetometers, thus 3D orientation of the node can be retrieved by combining these information. [5]

Finally, if we put one of these nodes on a surface, the node will provide data about the local tangency of the object at its location.

2.2 The surface

Our first MorphoShape is composed of nine sensors 3A3M (three axes of accelerometers, and 3 axes of magnetometers), distributed in a squared mesh (3 by 3), and distances between two neighbours is about 125mm. A linear circuit in "S" shape relies every node and is directly in the textile. The sensors are read via a SPI serial bus, which allows a lot of sensors. The serial bus is connected to a microcontroller controlled by a software driver, allowing to communicate with the surface, modifying some parameters and sequentially read all sensor values. The system sends the data to the host computer via a BlueTooth connection. In this version, the surface is connected for power supply, but a battery could be used. We can see the MorphoShape in Figure 1.

3. MORPHOSHAPE - RECONSTRUCTION ALGORITHMS

We have to acquire a surface via a set of sensors organised in a squared mesh laid on it. As it does not exist an intrinsic parameterization for surfaces (contrary to the curves with the arc-length parameter), we will keep the linear organization of sensors, thus the surface can be known by two families of curves in two complementary directions, where sensors are at the intersections, so that we have a tensorial topology of the surface. The figure 2 (a) shows a general description, with sensors equally distributed on a surface, and "virtual curves" are drawn as red ribbons (details on notations are given in next subsections).

We then have to reconstruct a surface from a finite number of curves laying on it. Methods have been developed to reconstruct curves via data from a ribbon (see Appendix A). Data are tangential data at sensors' positions, and distances between sensors along the curve. Let us notice that we do not

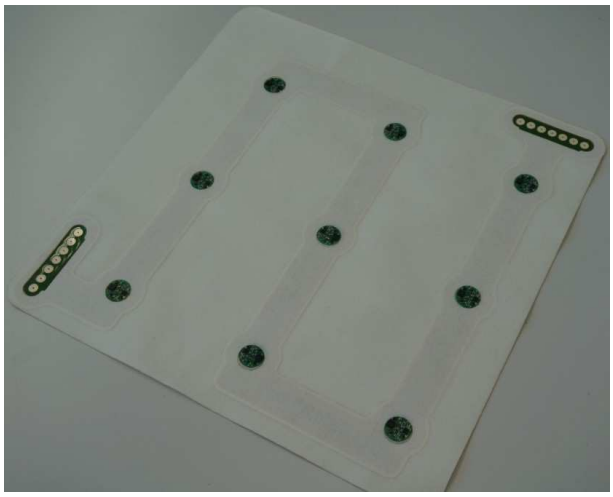
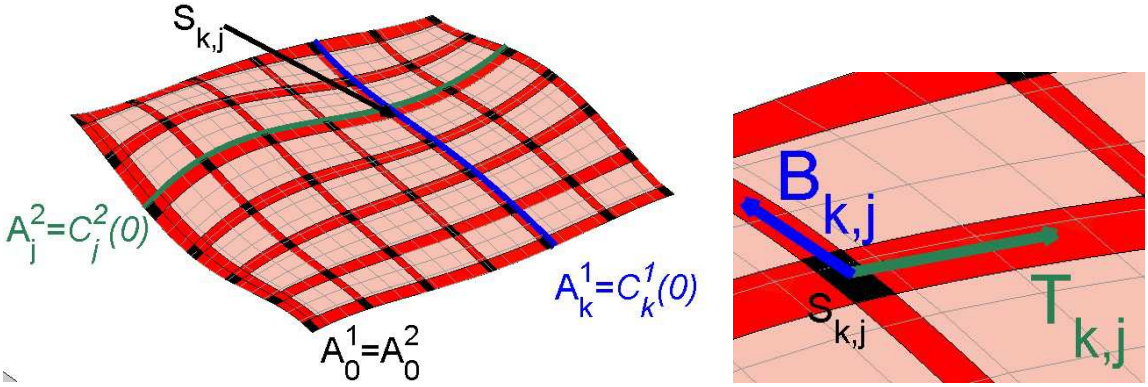


Figure 1 : Picture of MorphoShape, with the 9 sensor nodes

have any information concerning absolute positions of any points of the ribbon, such that the ribbon curves reconstructed are unique up to their starting point.

Hereinafter, we say curves are in two orthogonal directions, with curves named C_k^1 in the direction 1 (in blue on the figure), and C_j^2 in the direction 2 (in green).

In this section, the global methodology is detailed, with first how to obtain the tangential data from the sensors, then how to obtain the surface shape from the tangential data.



(a) General description of the surface and sensors squared distributed

(b) Sensor triaxal data

Figure 2 : MorphoShape model

3.1 From sensors to orientations

Each sensor's point $S_{k,j}$ (see Figure 2 (b)) provides its 3-D attitude, which is a matrix giving the coordinates of the reference frame linked to the sensor in an absolute reference. As curves are in two orthogonal directions, sensors have been placed such that we focus on two vectors from these matrices:

- $\mathbf{T_{k,j}}$ is the tangential vector of the curve j in the direction 2,
- $\mathbf{B_{k,j}}$ is the binormal vector of the curve j in the direction 2 in the tangent plane, which can also be seen as the tangential vector of the curve k in the direction 1.

In this subsection, the question is how to obtain tangential data from the sensors used (Accelerometers and Magnetometers).

This is done in two main stages : the first is to obtain physical values (said calibrated values) from sensors outputs; the second is to have the orientations from the calibrated values.

The first stage is then to obtain the transfer function of the sensor. Models for each modality are linear ones:

$$\underline{m_c} = \underline{\underline{S}}(\underline{m_r} - \underline{O_{ff}}) \tag{1}$$

where $\underline{m_r}$ are raw data, $\underline{m_c}$ are calibrated data, $\underline{\underline{S}}$ is the sensitivity matrix, defining scales of sensors, and non-orthogonalities, and $\underline{O_{ff}}$ is the offset vector. These parameters are defined during a calibration phasis (measurement campaigns and processings) .

As said previously, accelerometers measure the projection of the acceleration (couple of gravity and own acceleration) with additive noises :

$$m_{c,A}(k,j) = R_{k,j}.(g+a) + \varepsilon_a \tag{2}$$

In the same way, magnetometers measure the projection of the Earth Magnetic field, with additive noises :

$$m_{c,M}(k,j) = R_{k,j}.h + \varepsilon_h \tag{3}$$

with g, a and h written in an absolute reference. $R_{k,j}$ is then the matrix of the 3 sensors axes defined in the absolute reference.

Methods exist to find these matrices assuming no real perturbations (e.g. in [5]). We are also studying alternative methods able to filter signals when accelerations or magnetic perturbations occur, but we do not detail them here.

In the next step, the two first vectors in the matrices are used (vectors $\mathbf{T_{k,j}}$ and $\mathbf{B_{k,j}}$).

3.2 From orientations to the surface

Let us consider having $N+1$ sensors along the direction 1 and $n+1$ sensors along the direction 2. We then have data from sensors $S_{k,j}$ for $0 \leq k \leq n, 0 \leq j \leq N$. As the surface acquisition process induces a tensorial structure on the physical surface, we consider the following reconstruction strategy. At each time position:

1. we reconstruct the $(N+1)(n+1)$ 3D-curves from sensors data and length constraints,
2. these curves are adjusted according to the squared structure by translating their starting point,
3. then they curves are modified to create a closed mesh, because numerical computation and sensors errors make the mesh not strictly closed,
4. finally, the surface is filled by a standard cubic Coons process.

Let us detail some important steps of the previous process.

Step 1 - Independant curves reconstruction

For the direction 1, curves $C_k^1(s), 0 \leq k \leq n$ and s the arc-length parameter : $0 \leq s \leq L_k^1$ have to be reconstructed with the tangents $B_{k,j}, 0 \leq j \leq N$ and the arc-length parameters $s_{k,j}^1$ representing the length of the curve between the initial point and the sensor $S_{k,j}, 0 \leq j \leq N$, known by construction of the system.

In the same way, for the direction 2, curves $C_j^2(s), 0 \leq j \leq N, 0 \leq s \leq L_j^2$ have to be reconstructed with the tangents $T_{k,j}, 0 \leq k \leq n$ and the arc-length parameters $s_{k,j}^2$ representing the length of the curve between the initial point and the sensor $S_{k,j}, 0 \leq k \leq n$.

The method of reconstruction is detailed in Appendix A. As each curve is reconstructed up to an arbitrary starting point, we are now faced to adjust these curves according to the mesh.

Step 2 - Adjustment according to the mesh

This step is based on the two references curves, the first of each direction, which will fix the starting point of the others. Thus, the process is as following (refer to Figure 2).

- We first translate the curves $C_0^1(t)$ and $C_0^2(t)$ with the same starting point $C_0^1(0) = A_0^1 = A_0^2 = C_0^2(0)$
- The other starting points are defined via these 2 curves:
  - the $A_k^1, 1 \leq k \leq n$ are defined with $C_0^2(t)$ with $A_k^1 = C_0^2(s_{k,0}^2)$
  - the $A_j^2, 1 \leq j \leq N$ are defined with $C_0^1(t)$ with $A_j^2 = C_0^1(s_{0,j}^1)$

Step 3 - The curves network closure

At this stage, curves are well organised, but sensors precision and computational errors make the curves not exactly cross. This mesh has thus to be modified to enforce the closure. This process is an iterative one, modifying curves, piece by piece.

In order to simplify the explanations, let us split curves and define new notations. Let us define $C_k^1[j](t), 0 \leq t \leq 1$ the piece of curve $C_k^1(s)$ between the sensors $S_{k,j}$ and $S_{k,j+1}$, and in the same way $C_j^2[k](t), 0 \leq t \leq 1$ the piece of curve $C_j^2(s)$ between the sensors $S_{k,j}$ and $S_{k+1,j}$. (see Figure 3).

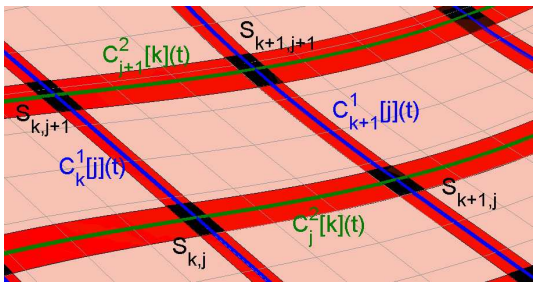


Figure 3 : Notations of curves per piece for mesh closure

Iteratively, the method deals with unitary patches. By construction, $C_k^1[j](t)$ and $C_j^2[k](t)$ have the same starting point.
Then, $C_{k+1}^1[j](t)$ will start at the end of $C_j^2[k](t)$ ($C_{k+1}^1[j](0) = C_j^2[k](1)$) and $C_{j+1}^2[k](t)$ will start at the end of $C_k^1[j](t)$ ($C_{j+1}^2[k](0) = C_k^1[j](1)$).

Thus, $C_{k+1}^1[j]$ and $C_{j+1}^2[k]$ will not necessary cross at their ending point (with previous considerations), we do have to modify these curves in order them to end at the same point. New curves will be noted $\tilde{C}_{k+1}^1[j]$ and $\tilde{C}_{j+1}^2[k]$ with the following constraints:

departure point position:

$\tilde{C}_{k+1}^1[j](0) = C_j^2[k](1)$ $\qquad$ $\tilde{C}_{j+1}^2[k](0) = C_k^1[j](1)$

departure tangent:

$\tilde{C}'^1_{k+1}[j](0) = B_{k+1,j}$ $\qquad$ $\tilde{C}'^1_{k+1}[j](1) = B_{k+1,j+1}$

ending tangent:

$\tilde{C}'^2_{j+1}[k](0) = T_{k,j+1}$ $\qquad$ $\tilde{C}'^2_{j+1}[k](1) = T_{k+1,j+1}$

length constraints:

$\int_0^1 \|C'^1_{k+1}[j](t)\|dt = (s^1_{k+1,j+1} - s^1_{k+1,j})$ $\qquad$ $\int_0^1 \|C'^2_{j+1}[k](t)\|dt = (s^2_{k+1,j+1} - s^2_{k,j+1})$

same ending point, unknown: $\tilde{C}_{k+1}^1[j](1) = \tilde{C}_{j+1}^2[k](1) = \mathbf{P}$

With these constraints, these segments are logically modeled as Hermite cubic curves (see Appendix B):

$$\tilde{C}^1_{k+1}[j](t) = \mathrm{H}[\tilde{C}^2_j[k](1), \mathbf{P}, B_{k+1,j}, B_{k+1,j+1}; 0, 1](t)$$

$$\tilde{C}^2_{j+1}[k](t) = \mathrm{H}[\tilde{C}^1_k[j](1), \mathbf{P}, T_{k,j+1}, T_{k+1,j+1}; 0, 1](t)$$

In order to find $\mathbf{P}$, we will use the two length constraints. But expanding these integrals yield non linear constraints, furthermore, these equations do not have a unique solution, we consider the supplementary energy minimization constraint:

$$\min\left((\int_0^1 \|C''^1_{k+1}[j](t)\|)^2 + (\int_0^1 \|C''^2_{j+1}[k](t)\|)^2 \right)$$

Finally, this minimisation under constraints allow to determine a unique final point, thus the curves are totally fixed.

Then, we translate the following pieces of the the curves $(C^1_{k+1}[j_2], j_2 > j)$ and $(C^2_{j+1}[k_2], k_2 > k)$ and we iterate.

Step 4 - The Surface filling

At this point, the physical surface is reconstructed/modeled by a network of two families of "orthogonal" spline curves meeting at sensors' position, delimiting thus a set of $n.N$ curvilinear rectangles.

Each of these curvilinear rectangles is then filled by a partially bi-cubically blended Coons process [6], producing a G^1 global surface.

4. SOFTWARE DEMONSTRATOR AND RESULTS

A software has been created to show the virtual surface defined from the data sensors. Precisely, the algorithms have been developed in Matlab, and integrated in a graphical user interface which communicates with the MorphoShape in real time. The interface displays at approximatively 15Hz the refreshed virtual surface. At each frame, the software reads sensors data, computes the surface with the algorithm described above, and displays it. As the surface is reconstructed up to its starting point, we suppose that it is a fixed point (the corner in the bottom left for this demonstation). Images in Figure 4 show some good visual results.

CONCLUSION

In this paper, we show how we have created a MorphoShape, from the physical surface to the graphical interface, via the algorithms, which is a surface equipped with sensors, able to provide its own shape and its deformations in real time. Several ways occur to go on in this domain. First, we can improve preprocessing to avoid accelerations from signals, and thus allowing dynamic motions. We can also characterize the performances of this with specific known shapes and comparing results. But the next real step is to combine the methodology to a case of Structural Health Monitoring, and to define a MorphoShape adapted to it (size, number of sensors, ...).

REFERENCES

[1] Sprynski N., Lacolle B., David D., and Biard L. Curve reconstruction via a ribbon of sensors. *Proceedings of the 14th IEEE International Conference on Electronics Circuits and Systems. ICECS, Marrakech, Morocco*, 12 2007.

[2] Huard M., Farouki R., Sprynski N., and Biard L. C2 interpolation of spatial data subject to arc length constraints using pythagorean quintic splines. *Graphical models*, 76, 2014.

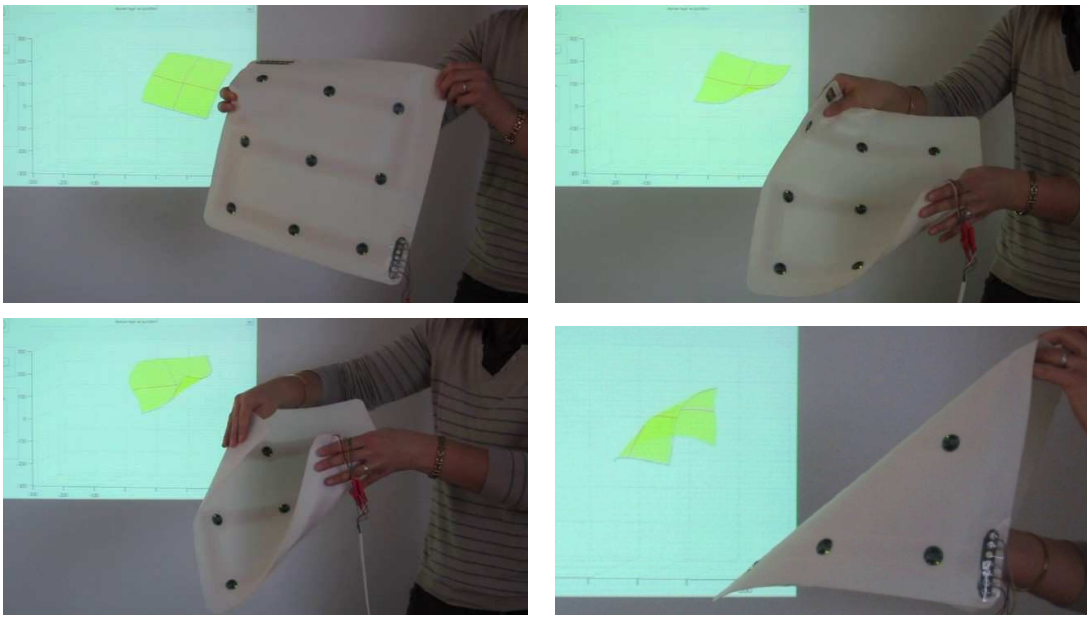


Figure 4 : Visual results: MorphoShape foreground, and its virtual surface in the background

[3] Sprynski N., Lacolle B., David D., and Biard L. Motion Capture of an Animated Surface via Sensors' ribbons, Surface Reconstruction via Tangential Measurements. *First International Conference on Pervasive and Embedded Computing and Communication Systems, PECCS 2011. Vilamoura, Algarve, Portugal*, pages 421–426, 3 2011.

[4] Huard M., Sprynski N., Szafran N., and Biard L. Reconstruction of quasi developable surfaces from ribbon curves. *Numerical algorithms*, 63:483506, 2012.

[5] Markley F.L. Quaternion Attitude estimation using vector observations. *Journal of the Astronautical Sciences*, 48:359–380, 2000.

[6] Coons S. *Surface patches and b-spline curves,Computer Aided Geometric Design*. In R. Barnhilland R. Riesenfeld editors, Academic Press, 1974.

[7] G.M. Nielson. ν-quaternion splines for the smooth interpolation of orientations. *IEEE Transactions on visualization and computer graphics*, 10(2):224–229, March 2004.

[8] G. Farin. *Curves and Surfaces for CAGD - Fifth Edition*. Academic Press, 2002.

APPENDIX

A - Initial Reconstructed Curves – A 3D curve $C(s) = (x(s), y(s), z(s))$, parameterized with respect to its arc-length s satisfy $|C'(s)| \equiv 1$, so that the derivative curve $C'(s)$ is a curve lying on the unit sphere. Initial data are unit tangential vectors at points with assigned arc length parameters. The methodology is thus as follows.

- First, we interpolate data using cubic splines on the sphere, leading to the derivative curve $C'(s)$.
- Then, by integration we get a solution for $C(s)$.

Cubic splines on the unit sphere (see [7]) are an extension of the usual B-splines in the euclidian space. The main differences are the following.
$\diamond$ The evaluation of the control polygon of cubic splines on the spherical space requires to solve a non linear system through an iterative algorithm.
$\diamond$ The usual De Casteljau algorithm, based on linear interpolations, has to be replaced by the *spherical interpolation*

$$Slerp(\mathbf{a}, \mathbf{b}, t) = \frac{\sin((1-t)\theta)\mathbf{a} + \sin(t\theta)\mathbf{b}}{\sin(\theta)},$$

where **a** and **b** are two unit vectors, θ the angle between vectors **a** and **b**, and $t \in [0,1]$.

It is proved in [1] that this construction is invariant under rotations and scaling, and that these spherical splines minimize a combination of the curvature κ_1, the torsion κ_2, and the variations of the curvature, precisely

$$\min \int (\kappa_1'^2 + \kappa_1^2 (\kappa_1'^2 + \kappa_2^2)),$$

which gives physical sense to the reconstruction.

B - Cubic Hermite Interpolation – Given spatial points $\mathbf{p}_0$ and $\mathbf{p}_1$ associated with tangent vectors $\mathbf{t}_0$ and $\mathbf{t}_1$, together with two parameters α_0 and α_1 $(\alpha_0 < \alpha_1)$, there exists a unique cubic spatial parametric curve $\mathbf{r}(t)$ such that

$$\mathbf{r}(\alpha_0) = \mathbf{p}_0\,,\ \mathbf{r}(\alpha_1) = \mathbf{p}_1\,,\ \mathbf{r}'(\alpha_0) = \mathbf{t}_0\,,\ \mathbf{r}'(\alpha_1) = \mathbf{t}_1\,.$$

Precisely, $\mathbf{r}(t)$ is defined by

$$\begin{aligned}\mathbf{r}(t) = {} & H_0(\hat{t})\,\mathbf{p}_0 + H_1(\hat{t})\,\mathbf{p}_1 \\ & + (\alpha_1 - \alpha_0) H_2(\hat{t})\,\mathbf{t}_0 + (\alpha_1 - \alpha_0) H_3(\hat{t})\,\mathbf{t}_1\,,\end{aligned}$$

with $\hat{t} = \frac{t-\alpha_0}{\alpha_1-\alpha_0}$ and where functions φ_j are the cubic Hermite polynomials [8] on the interval $[0,1]$, and $\mathbf{r}(t)$ and is denoted shortly by

$$\mathbf{r}(t) = \mathrm{H}[\mathbf{p}_0, \mathbf{p}_1, \mathbf{t}_0, \mathbf{t}_1; \alpha_0, \alpha_1](t)\,.$$