Draft for G-DIC with CS volumique FE

E. Andó E. Roubin J.L. Cercos-Pita

Last update : 28th June 2019

1 Tetrahedron shape function

1.1 Notations

- $\boldsymbol{x} = [x \ y \ z]^T$: Cartesian coordinates
- $\boldsymbol{x}_{i,k} = [x_{i,k} \ y_{i,k} \ z_{i,k}]^T$ with $k = \{1..4\}$: nodes coordinates for an arbitrary i-th tetrahedron.
- $L_{i,k}(\boldsymbol{x})$ with $k = \{1..4\}$: volume coordinates of the arbitrary i-th tetrahedron
- $N_{i,k}(\boldsymbol{x})$ with $k = \{1..4\}$: shape functions of the arbitrary i-th tetrahedron
- V_i : volume of the arbitrary i-th tetrahedron

1.2 Volume coordinates

We introduce the special coordinates:

$$\begin{cases} x = & L_{i,1}x_{i,1} + L_{i,2}x_{i,2} + L_{i,3}x_{i,3} + L_{i,4}x_{i,4} \\ y = & L_{i,1}y_{i,1} + L_{i,2}y_{i,2} + L_{i,3}y_{i,3} + L_{i,4}y_{i,4} \\ z = & L_{i,1}z_{i,1} + L_{i,2}z_{i,2} + L_{i,3}z_{i,3} + L_{i,4}z_{i,4} \\ 1 = & L_{i,1} + L_{i,2} + L_{i,3} + L_{i,4} \end{cases} \tag{1}$$

Solving (1) gives the volume coordinates:

$$L_{i,k} = \frac{a_{i,k} + b_{i,k}x + c_{i,k}y + d_{i,k}z}{6V}, \ k = \{1..4\} \tag{2}$$

with:

$$6V = \begin{vmatrix} 1 & x_{i,1} & y_{i,1} & z_{i,1} \\ 1 & x_{i,2} & y_{i,2} & z_{i,2} \\ 1 & x_{i,3} & y_{i,3} & z_{i,3} \\ 1 & x_{i,4} & y_{i,4} & z_{i,4} \end{vmatrix} \tag{3}$$

and

$$a_{i,1} = \begin{vmatrix} x_{i,2} & y_{i,2} & z_{i,2} \\ x_{i,3} & y_{i,3} & z_{i,3} \\ x_{i,4} & y_{i,4} & z_{i,4} \end{vmatrix}, \ b_{i,1} = -\begin{vmatrix} 1 & y_{i,2} & z_{i,2} \\ 1 & y_{i,3} & z_{i,3} \\ 1 & y_{i,4} & z_{i,4} \end{vmatrix}, \ c_{i,1} = \begin{vmatrix} 1 & x_{i,2} & z_{i,2} \\ 1 & x_{i,3} & z_{i,3} \\ 1 & x_{i,4} & z_{i,4} \end{vmatrix}, \ d_{i,1} = -\begin{vmatrix} 1 & x_{i,2} & y_{i,2} \\ 1 & x_{i,3} & y_{i,3} \\ 1 & x_{i,4} & y_{i,4} \end{vmatrix}$$

$$a_{i,2} = -\begin{vmatrix} x_{i,1} & y_{i,1} & z_{i,1} \\ x_{i,3} & y_{i,3} & z_{i,3} \\ x_{i,4} & y_{i,4} & z_{i,4} \end{vmatrix},\ b_{i,2} = \begin{vmatrix} 1 & y_{i,1} & z_{i,1} \\ 1 & y_{i,3} & z_{i,3} \\ 1 & y_{i,4} & z_{i,4} \end{vmatrix},\ c_{i,2} = -\begin{vmatrix} 1 & x_{i,1} & z_{i,1} \\ 1 & x_{i,3} & z_{i,3} \\ 1 & x_{i,4} & z_{i,4} \end{vmatrix},\ d_{i,2} = \begin{vmatrix} 1 & x_{i,1} & y_{i,1} \\ 1 & x_{i,3} & y_{i,3} \\ 1 & x_{i,4} & y_{i,4} \end{vmatrix}$$

$$a_{i,3} = \begin{vmatrix} x_{i,1} & y_{i,1} & z_{i,1} \\ x_{i,2} & y_{i,2} & z_{i,2} \\ x_{i,4} & y_{i,4} & z_{i,4} \end{vmatrix},\ b_{i,3} = -\begin{vmatrix} 1 & y_{i,1} & z_{i,1} \\ 1 & y_{i,2} & z_{i,2} \\ 1 & y_{i,4} & z_{i,4} \end{vmatrix},\ c_{i,3} = \begin{vmatrix} 1 & x_{i,1} & z_{i,1} \\ 1 & x_{i,2} & z_{i,2} \\ 1 & x_{i,4} & z_{i,4} \end{vmatrix},\ d_{i,3} = -\begin{vmatrix} 1 & x_{i,1} & y_{i,1} \\ 1 & x_{i,2} & y_{i,2} \\ 1 & x_{i,4} & y_{i,4} \end{vmatrix}$$

$$a_{i,4} = -\begin{vmatrix} x_{i,1} & y_{i,1} & z_{i,1} \\ x_{i,2} & y_{i,2} & z_{i,2} \\ x_{i,3} & y_{i,3} & z_{i,3} \end{vmatrix},\ b_{i,4} = \begin{vmatrix} 1 & y_{i,1} & z_{i,1} \\ 1 & y_{i,2} & z_{i,2} \\ 1 & y_{i,3} & z_{i,3} \end{vmatrix},\ c_{i,4} = -\begin{vmatrix} 1 & x_{i,1} & z_{i,1} \\ 1 & x_{i,2} & z_{i,2} \\ 1 & x_{i,3} & z_{i,3} \end{vmatrix},\ d_{i,4} = \begin{vmatrix} 1 & x_{i,1} & y_{i,1} \\ 1 & x_{i,2} & y_{i,2} \\ 1 & x_{i,3} & y_{i,3} \end{vmatrix}$$

Fun fact:

$$L_{i,1} = \frac{\text{volume}(P234)}{V},\ L_{i,2} = \frac{\text{volume}(P134)}{V},\ \ldots \tag{4}$$

1.3 Shape functions

We simply have

$$N_{i,k}(\boldsymbol{x}) = L_{i,k}(\boldsymbol{x}) \tag{5}$$

2 Global Digital Image Correlation

2.1 Formulation of the problem

2.1.1 Notations

An image is define as a scalar function that gives gray level at each discrete point of space.

- $\boldsymbol{x} = [x\ y\ z]^T$: spatial coordinates
- $f(\boldsymbol{x})$: reference image
- $\boldsymbol{\nabla} f(\boldsymbol{x}) = [f_x(\boldsymbol{x})\ f_y(\boldsymbol{x})\ f_z(\boldsymbol{x})]^T$: gradient of the reference image
- $g(\boldsymbol{x})$: deformed image
- $\boldsymbol{u}(\boldsymbol{x}) = [u_x(\boldsymbol{x})\ u_y(\boldsymbol{x})\ u_z(\boldsymbol{x})]^T$: displacement field

2.1.2 Strong form

The reference and the deformed images can be related through the displacement field by:

$$g(\boldsymbol{x}) = f(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) \tag{6}$$

which requires the conservation of the optical flow.

Assuming that the reference image is differentiable (Important: see Besnard, Hild and Roux 2006), a Taylor expansion to the first order can be yield:

$$g(\boldsymbol{x}) = f(\boldsymbol{x}) + \boldsymbol{u}(\boldsymbol{x}) \cdot \boldsymbol{\nabla} f(\boldsymbol{x}) + \eta, \tag{7}$$

where η is the error to become minimized.

2.1.3 Weak form

To estimate $\boldsymbol{u}(\boldsymbol{x})$ the quadratic difference between left and right members of the strong formulation is integrated over the studied domain Ω and subsequently minimized:

$$\eta^2 = \int_\Omega \left[\boldsymbol{u}(\boldsymbol{x}) \cdot \boldsymbol{\nabla} f(\boldsymbol{x}) + f(\boldsymbol{x}) - g(\boldsymbol{x})\right]^2 d\Omega \tag{8}$$

(Actually your are making 2 different steps: First, you are accepting an error in Eq. 7. Second you are squaring and averaging the error.)

2.2 FE discretization

The support is defined by the set of n_{el} FE and n nodes. This allows to split the integral of the error by a sum of integrals of each tetrahedron,

$$\eta^2 = \sum_{i=1}^{n_{\mathrm{el}}} \eta_i^2 = \sum_{i=1}^{n_{\mathrm{el}}} \int_{\Omega_i} \left[\boldsymbol{u}(\boldsymbol{x}) \cdot \boldsymbol{\nabla} f(\boldsymbol{x}) + f(\boldsymbol{x}) - g(\boldsymbol{x})\right]^2 d\Omega. \tag{9}$$

Please, notice that no approximation has been practiced yet. Indeed, the approximation comes from the consideration of constant magnitudes among the FE,

$$\eta^2 \simeq \sum_{i=1}^{n_{\mathrm{el}}} \left[\boldsymbol{u}(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i) + \langle f - g \rangle (\boldsymbol{x}_i)\right]^2 V_i, \tag{10}$$

where $\langle \cdot \rangle$ denotes a convolution, to be defined yet. i.e. the closest pixel to $\boldsymbol{x}_i$, the average along the whole FE, an actual convolution from the image data...

2.2.1 Discretized displacement

The FE displacement $\boldsymbol{u}(\boldsymbol{x}_i)$ is computed from the displacements at the finite element nodes, $\boldsymbol{d}_a = \left[d_{a_x} \; d_{a_y} \; d_{a_z}\right]^T$ and the shape functions $N_a(\boldsymbol{x}_i)$. The number of degree of freedom is $n_{\mathrm{dof}} = 3 \times n$.

The displacement field is discretized as follow:

$$\boldsymbol{u}(\boldsymbol{x}_i) = \sum_{a=1}^{n} N_a(\boldsymbol{x}_i)\boldsymbol{d}_a = \boldsymbol{N}(\boldsymbol{x}_i) \cdot \boldsymbol{d} \tag{11}$$

with:

$$\underbrace{\boldsymbol{N}(\boldsymbol{x}_i)}_{3\times n_{\mathrm{dof}}} = \begin{bmatrix} N_1(\boldsymbol{x}_i) & 0 & 0 & N_2(\boldsymbol{x}_i) & 0 & 0 & \dots & N_n(\boldsymbol{x}_i) & 0 & 0 \\ 0 & N_1(\boldsymbol{x}_i) & 0 & 0 & N_2(\boldsymbol{x}_i) & 0 & \dots & 0 & N_n(\boldsymbol{x}_i) & 0 \\ 0 & 0 & N_1(\boldsymbol{x}_i) & 0 & 0 & N_2(\boldsymbol{x}_i) & \dots & 0 & 0 & N_n(\boldsymbol{x}_i) \end{bmatrix} \tag{12}$$

and

$$\underbrace{\boldsymbol{d}}_{n_{\mathrm{dof}}\times 1} = \begin{bmatrix} d_{1_x} & d_{1_y} & d_{1_z} & d_{2_x} & d_{2_y} & d_{2_z} & \dots & d_{n_x} & d_{n_y} & d_{n_z} \end{bmatrix}^T \tag{13}$$

Here an abuse of notation has been practised. Indeed, $N_a(\boldsymbol{x}_i) = N_{i,k}(\boldsymbol{x}_i)$, are the shape functions associated to the i-th FE, which contains the point $\boldsymbol{x}_i$. Along this line, all the shape functions associated to nodes that don't belongs to the i-th FE are null.

2.2.2 FE formulation

The discretized version of the functional η can be written:

$$\eta^2 = \sum_{i=1}^{n_{\mathrm{el}}} \left[\boldsymbol{N}(\boldsymbol{x}_i) \cdot \boldsymbol{d} \cdot \langle \boldsymbol{\nabla} f(\boldsymbol{x}) \rangle + \langle f - g(\boldsymbol{x}) \rangle\right]^2 V_i. \tag{14}$$

Minimizing η^2 leads to the resolution of:

$$\begin{aligned} \underbrace{\frac{\partial \eta^2}{\partial \boldsymbol{d}}}_{n_{\mathrm{dof}}\times 1} &= 2\sum_{i=1}^{n_{\mathrm{el}}} \left[\boldsymbol{N}(\boldsymbol{x}_i) \cdot \boldsymbol{d} \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i) + \langle f - g \rangle (\boldsymbol{x}_i)\right] \left[\boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i)\right] V_i \\ &= 2\sum_{i=1}^{n_{\mathrm{el}}} \left[\boldsymbol{N}(\boldsymbol{x}_i) \cdot \boldsymbol{d} \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i)\right] \left[\boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i)\right] V_i + \langle f - g \rangle (\boldsymbol{x}_i) \boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i) V_i \\ &= 2\left(\sum_{i=1}^{n_{\mathrm{el}}} \left[\boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i)\right] \cdot \left[\langle \boldsymbol{\nabla} f \rangle^T (\boldsymbol{x}_i) \cdot \boldsymbol{N}(\boldsymbol{x}_i)\right] V_i \, \boldsymbol{d} + \sum_{i=1}^{n_{\mathrm{el}}} \langle f - g \rangle (\boldsymbol{x}_i) \boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i) V_i\right) \\ &= \boldsymbol{0} \end{aligned} \tag{15}$$

Finally, the system to solve is:

$$\boldsymbol{M}\boldsymbol{d} = \boldsymbol{F} \tag{16}$$

with the "stiffness" matrix

$$\underbrace{\boldsymbol{M}}_{n_{\mathrm{dof}} \times n_{\mathrm{dof}}} = \sum_{i=1}^{n_{\mathrm{el}}} \left[\boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i)\right] \cdot \left[\langle \boldsymbol{\nabla} f \rangle^T (\boldsymbol{x}_i) \cdot \boldsymbol{N}(\boldsymbol{x}_i)\right] \tag{17}$$

and the right hand side vector:

$$\underbrace{\boldsymbol{F}}_{n_{\mathrm{dof}} \times 1} = \sum_{i=1}^{n_{\mathrm{el}}} \langle f - g \rangle (\boldsymbol{x}_i) \boldsymbol{N}^T(\boldsymbol{x}_i) \cdot \langle \boldsymbol{\nabla} f \rangle (\boldsymbol{x}_i). \tag{18}$$

2.2.3 Elementary stiffness matrix

The elementary stiffness matrix have the following general expression:

$$M^e_{[\alpha a, \beta b]} = \int_{\Omega_e} N_a(\boldsymbol{x}) N_b(\boldsymbol{x}) f_\alpha(\boldsymbol{x}) f_\beta(\boldsymbol{x}) d\Omega \tag{19}$$

with α and β being iterators for the spatial direction (x, y, z) and a and b being iterators for the nodes $(1, 2, 3, 4)$ (in their local numerotation).

It of size 12×12 (number of nodes $\times$ number of spatial dimensions) can be decomposed in 4×4 (number of nodes) sub-matrices $\boldsymbol{M}^e_{ab}$ of size 3×3 (number of spatial dimensions) as follows:

$$\boldsymbol{M}^e = \begin{bmatrix} \boldsymbol{M}^e_{11} & \boldsymbol{M}^e_{12} & \boldsymbol{M}^e_{13} & \boldsymbol{M}^e_{14} \\ \text{sym} & \boldsymbol{M}^e_{22} & \boldsymbol{M}^e_{23} & \boldsymbol{M}^e_{24} \\ \text{sym} & \text{sym} & \boldsymbol{M}^e_{33} & \boldsymbol{M}^e_{34} \\ \text{sym} & \text{sym} & \text{sym} & \boldsymbol{M}^e_{44} \end{bmatrix} \tag{20}$$

where

$$\boldsymbol{M}^e_{ab} = \int_{\Omega_e} N_a(\boldsymbol{x}) N_b(\boldsymbol{x}) \underbrace{\begin{bmatrix} f_x(\boldsymbol{x})^2 & f_x(\boldsymbol{x}) f_y(\boldsymbol{x}) & f_x(\boldsymbol{x}) f_z(\boldsymbol{x}) \\ \text{sym} & f_y^2(\boldsymbol{x}) & f_y(\boldsymbol{x}) f_z(\boldsymbol{x}) \\ \text{sym} & \text{sym} & f_z^2(\boldsymbol{x}) \end{bmatrix}}_{\boldsymbol{\nabla} f(\boldsymbol{x}) \otimes \boldsymbol{\nabla} f(\boldsymbol{x})} d\Omega \tag{21}$$

2.2.4 Elementary second hand vector

The elementary second hand vector have the following general expression:

$$F^e_{\alpha a} = \int_{\Omega_e} (g(\boldsymbol{x}) - f(\boldsymbol{x})) N_a(\boldsymbol{x}) f_\alpha(\boldsymbol{x}) d\Omega \tag{22}$$

with α being an iterator for the spatial direction (x, y, z) and a being an iterator for the nodes $(1, 2, 3, 4)$ (in their local numerotation).

It is of size 12×1 (number of nodes $\times$ number of spatial dimensions) can be decomposed in 4×1 (number of nodes) sub-vectors $\boldsymbol{F}^e_{ab}$ of size 3×1 (number of spatial dimensions) as follows:

$$\boldsymbol{F}^e = \begin{bmatrix} \boldsymbol{F}^e_1 \\ \boldsymbol{F}^e_2 \\ \boldsymbol{F}^e_3 \\ \boldsymbol{F}^e_4 \end{bmatrix} \tag{23}$$

where

$$\boldsymbol{F}^e_a = \int_{\Omega_e} \left(f(\boldsymbol{x}) - g(\boldsymbol{x})\right) N_a(\boldsymbol{x}) \underbrace{\begin{bmatrix} f_x(\boldsymbol{x}) \\ f_y(\boldsymbol{x}) \\ f_z(\boldsymbol{x}) \end{bmatrix}}_{\boldsymbol{\nabla} f(\boldsymbol{x})} d\Omega \tag{24}$$

2.3 The role of the image gradient

From Eq. 7 it is quite evident that the image gradient plays a main role in the linear system arranged afterwards. Herein it is demonstrated that null gradient components lead to singular matrices. To this end, we focus on an arbitrary point, $\boldsymbol{x}_A$, associated to an arbitrary A-th mesh node. The linear system matrix should then takes the form

$$\boldsymbol{M} = \sum_{i=1}^{n_{\mathrm{el}}} \left(\begin{array}{ccccccc} & & & \cdots & & & \\ & & N_J N_{J-1} f_x^2 & N_J N_{J-1} f_x f_y & N_J N_{J-1} f_x f_z & & \\ & N_{J-1} N_J f_x f_z & N_J^2 f_x^2 & N_J^2 f_x f_y & N_J^2 f_x f_z & N_{J+1} N_J f_z f_x & \\ \cdots & N_{J-1} N_J f_y f_z & N_J^2 f_y f_x & N_J^2 f_y^2 & N_J^2 f_y f_z & N_{J+1} N_J f_y f_x & \cdots \\ & N_{J-1} N_J f_z^2 & N_J^2 f_z f_x & N_J^2 f_z f_y & N_J^2 f_z^2 & N_{J+1} N_J f_z f_x & \\ & & N_{J+1} N_J f_x^2 & N_{J+1} N_J f_x f_y & N_{J+1} N_J f_x f_z & & \\ & & & \cdots & & & \end{array} \right), \tag{25}$$

where both, the shape functions and the image derivatives, are evaluated at $\boldsymbol{x}_i$.

Now let's define a characteristic length of the mesh, h, which features the distance between nodes in the mesh, such that we can assert

$$\boldsymbol{\nabla} f(\boldsymbol{x}_i) = \boldsymbol{\nabla} f(\boldsymbol{x}_A) + O(h), \tag{26}$$

with h decreasing when the mesh resolution is improved. Thus, if a component of the gradient is null, $f_x(\boldsymbol{x}_A) = 0$ for instance, then the matrix takes the form:

$$\boldsymbol{M} = \sum_{i=1}^{n_{\mathrm{el}}} \left(\begin{array}{ccccccc} & & & \cdots & & & \\ & & O(h) & N_J N_{J-1} f_x f_y & N_J N_{J-1} f_x f_z & & \\ & O(h) & O(h) & O(h) & O(h) & O(h) & \\ \cdots & N_{J-1} N_J f_y f_z & O(h) & N_J^2 f_y^2 & N_J^2 f_y f_z & N_{J+1} N_J f_y f_x & \cdots \\ & N_{J-1} N_J f_z^2 & O(h) & N_J^2 f_z f_y & N_J^2 f_z^2 & N_{J+1} N_J f_z f_x & \\ & & O(h) & N_J N_{J+1} f_x f_y & N_J N_{J+1} f_x f_z & & \\ & & & \cdots & & & \end{array} \right), \tag{27}$$

which becomes singular in the continuous, $h \to 0$. In practice, the matrix becomes singular in the very first moment a component of the gradient vanish within all the mesh elements that own a common node.

2.3.1 Meaning of the gradient-lead singular behavior

Actually this singular behavior is not unique of G-DIC, but happens in all methods without a regularization factor. That's because explicitly minimizing Eq. 7 actually leads to infinite solutions,

$$\eta = 0 \rightarrow \boldsymbol{u}(\boldsymbol{x}) = (g(\boldsymbol{x}) - f(\boldsymbol{x}))\frac{\boldsymbol{\nabla} f(\boldsymbol{x})}{|\boldsymbol{\nabla} f(\boldsymbol{x})|^2} + k\boldsymbol{n}(\boldsymbol{x}), \tag{28}$$

with $\boldsymbol{n}(\boldsymbol{x}) \cdot \boldsymbol{\nabla} f(\boldsymbol{x}) = 0$ and k an arbitrary constant. In many methods $k = 0$ is selected.

The singular behavior described above is nothing else but a situation where $\boldsymbol{n}$ is aligned with an axis, x, y or z. Possible solutions are:

1. Applying a homogeneous Dirichlet BC, i.e. $\boldsymbol{d}_A \cdot \boldsymbol{n}_A = 0$
2. Applying a homogeneous Neumann BC, i.e. $\frac{\partial \boldsymbol{d}_A}{\partial \boldsymbol{n}_A} = 0$
3. Introducing a regularization term

3 Test

3.1 Displacement and Rotation limits Mickey Mouse

Image = 256^3
Correlation Length = 51.2px
Border = 26px
At 15px rigid displacement, fail, at 15° also (34 pixels at the edge)

3.2 Displacement and Rotation limits Serious (but without Dynamics – strange!)

Going to try a number of random fields with different correlation lengths.
In ER's mind: One random field realisation is 1x1x1 bananas (800x800x800 nodes).
Gaussian Variance set to centred (on zero) Gaussian distribution, giving more or less a spread of ±3 greylevels.
Turning it into a 16b image between ±4.
Correlation lengths vary: Bananas: {1.00, 0.50, 0.10, 0.05} Pixels: {800, 400, 80, 40}
We make variations of:

- l: correlation length
- b: border size [px]
- $\boldsymbol{t} = [t_x\ t_y\ t_z]$: translation vector [px]
- $\boldsymbol{r} = [r_x\ r_y\ r_z]$: rotation vector [deg]
- d: Mesh density
- n: Number of iteration before convergency (0.1%)

l	b	$\boldsymbol{t}$	$\boldsymbol{r}$	d	n
80	25	[5 0 0]	[0 0 0]	50	4
80	25	[10 0 0]	[0 0 0]	50	3
80	25	[15 0 0]	[0 0 0]	50	4
80	25	[20 0 0]	[0 0 0]	50	(5%)
80	50	[20 0 0]	[0 0 0]	50	6
40	25	[20 0 0]	[0 0 0]	50	(6%)

3.3 Displacements limits (Very Serious)

Random fields distribution: Standard normal distribution (Gaussian with zero mean and unit variance)
Random fields covariance function: Gaussian correlation function with correlation length l

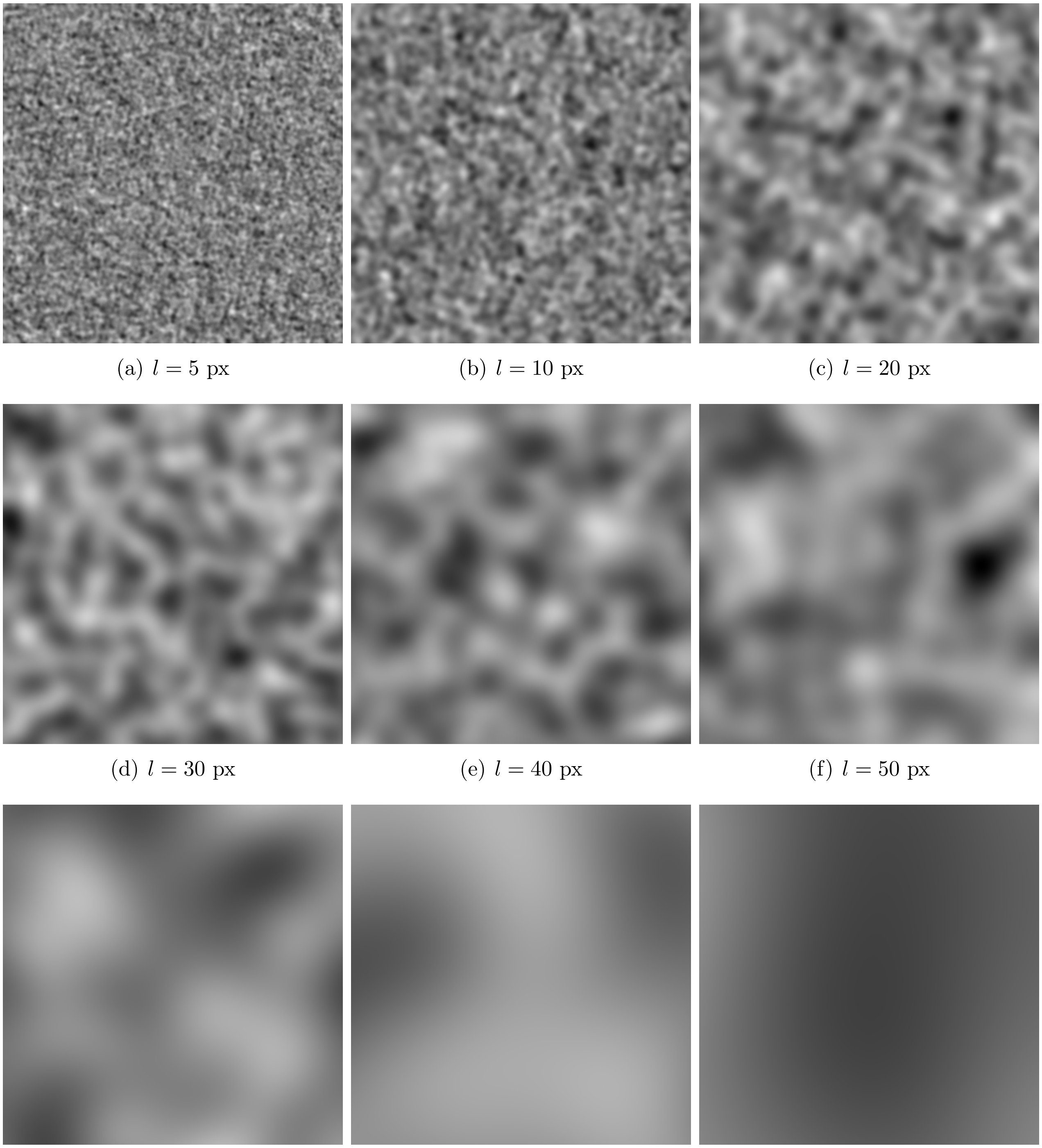
The correlated Random Fields are defined over a cube of size 1 with correlation lengths of:

$$l = \{0.01\ 0.02\ 0.04\ 0.06\ 0.08\ 0.1\ 0.2\ 0.4\ 0.6\}$$

and realisations are discretised over $500 \times 500 \times 500$ pixels cubes which corresponds to correlations lengths of

$$l = \{5\ 10\ 20\ 30\ 40\ 50\ 100\ 200\ 300\}\ \text{px}$$

l	b	$\boldsymbol{t}$	$\boldsymbol{r}$	d	n
5	30	[1 0 0]	[0 0 0]	50	1
5	30	[2 0 0]	[0 0 0]	50	3
5	30	[3 0 0]	[0 0 0]	50	6
5	30	[4 0 0]	[0 0 0]	50	6
5	30	[5 0 0]	[0 0 0]	50	6
5	30	[6 0 0]	[0 0 0]	50	div
10	30	[1 0 0]	[0 0 0]	50	1
10	30	[3 0 0]	[0 0 0]	50	2
10	30	[2 0 0]	[0 0 0]	50	2
10	30	[4 0 0]	[0 0 0]	50	3
10	30	[5 0 0]	[0 0 0]	50	5
10	30	[6 0 0]	[0 0 0]	50	10
10	30	[7 0 0]	[0 0 0]	50	20
10	30	[8 0 0]	[0 0 0]	50	div
20	30	[1 0 0]	[0 0 0]	50	1
20	30	[3 0 0]	[0 0 0]	50	2
20	30	[2 0 0]	[0 0 0]	50	2
20	30	[4 0 0]	[0 0 0]	50	2
20	30	[5 0 0]	[0 0 0]	50	2
20	30	[6 0 0]	[0 0 0]	50	3
20	30	[7 0 0]	[0 0 0]	50	3
20	30	[8 0 0]	[0 0 0]	50	??
20	30	[9 0 0]	[0 0 0]	50	5
20	30	[10 0 0]	[0 0 0]	50	7
20	30	[11 0 0]	[0 0 0]	50	11 (div at 0.2)



(a) $l = 5$ px (b) $l = 10$ px (c) $l = 20$ px

(d) $l = 30$ px (e) $l = 40$ px (f) $l = 50$ px

(g) $l = 100$ px (h) $l = 200$ px (i) $l = 300$ px

Figure 1: Slices of $500 \times 500 \times 500$ vx realisations with various correlation lengths

l	d	b	Max t_x	n	commentaire
5	30	30	3	5	conv lentement
10	30	30	5	4	
20	30	30			
30	30	30			
40	30	30			
50	30	30			
100	30	30			
200	30	30			
300	30	30			