INSTITUT CLÉMENT ADER

Semaine de GTs, MECAMAT
June 13th, 2023, Online

Algoritmic view of **DIC outside small transformations**

J.-C. Passieux
J.N. Périé and R. Bouclier and J.E. Dufour and R. Fouque and A. Rouwane

Institut Clément Ader (ICA), Université de Toulouse, CNRS/INSA/Mines Albi/UPS/ISAE, Toulouse, France

IMT Mines Albi-Carmaux
École Mines-Télécom
INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE
Isae Institut Supérieur de l'Aéronautique et de l'Espace SUPAERO
UNIVERSITÉ TOULOUSE III PAUL SABATIER
cnrs
Université Fédérale Toulouse Midi-Pyrénées

Many issues...

The talk will mainly focus on **Formulation** and **Solution algorithms**.

1. Solver
2. Initialization
3. Regularisation
4. Occultation + light effects + pixel size

$\rightarrow$ Towards a new photometric formulation

This talk won't consider patterning problems and other experimentally related issues

1. SOLVER

FE-DIC: Algebraic setting

- **Gray level residual $\mathbf{r} \in \mathbb{R}^m$** [Horn & Schunk 81] :

$$\mathbf{r}(\mathbf{x}, \mathbf{u}) = \mathbf{f}(\mathbf{x}) - \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}).$$

m number of pixels or quadrature points.

FE-DIC: Algebraic setting

- **Gray level residual $\mathbf{r} \in \mathbb{R}^m$** [Horn & Schunk 81] :

$$\mathbf{r}(\mathbf{x}, \mathbf{u}) = \mathbf{f}(\mathbf{x}) - \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}).$$

  m number of pixels or quadrature points.

- **Unknown transformation $\phi(\mathbf{x}, \mathbf{u})$**

$$\phi(\mathbf{x}, \mathbf{u}) = \mathbf{x} + \mathbf{N}(\mathbf{x})\ \mathbf{u}$$

  $\mathbf{N}(\mathbf{x})$ matrix of basis functions, $\mathbf{u} \in \mathbb{R}^n$ DOF vector, n: number of DOF

FE-DIC: Algebraic setting

- **Gray level residual $\mathbf{r} \in \mathbb{R}^m$** [Horn & Schunk 81] :

$$\mathbf{r}(\mathbf{x}, \mathbf{u}) = \mathbf{f}(\mathbf{x}) - \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}).$$

m number of pixels or quadrature points.

- **Unknown transformation $\phi(\mathbf{x}, \mathbf{u})$**

$$\phi(\mathbf{x}, \mathbf{u}) = \mathbf{x} + \mathbf{N}(\mathbf{x})\ \mathbf{u}$$

$\mathbf{N}(\mathbf{x})$ matrix of basis functions, $\mathbf{u} \in \mathbb{R}^n$ DOF vector, n: number of DOF

- An **unconstrained non-linear least square** problem:

$$\boxed{\mathbf{u}^\star = \arg\min_{\mathbb{R}^n} \quad j(\mathbf{u})} \qquad \text{with} \qquad j(\mathbf{u}) = \frac{1}{2}\, \mathbf{r}(\mathbf{x}, \mathbf{u})^T \mathbf{r}(\mathbf{x}, \mathbf{u})$$

What is a descent direction?

- Principle of **descent algorithms**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \alpha \cdot \mathbf{d}$$

  - **d**: search direction (many choices: next slide)
  - α: step size (line search)

What is a descent direction?

- Principle of **descent algorithms**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \alpha \cdot \mathbf{d}$$

  - **d**: search direction (many choices: next slide)
  - α: step size (line search)

- important **convergence condition**: **d** a descent direction iff:

$$j(\mathbf{u}^k) > j(\mathbf{u}^{k+1}) \quad \Rightarrow \quad \boxed{\boldsymbol{\nabla} j^{(k)\,T}\,\mathbf{d} < 0}$$

What is a descent direction?

- Principle of **descent algorithms**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \alpha \cdot \mathbf{d}$$

  - **d**: search direction (many choices: next slide)
  - α: step size (line search)

- important **convergence condition**: **d** a descent direction iff:

$$j(\mathbf{u}^k) > j(\mathbf{u}^{k+1}) \quad \Rightarrow \quad \boxed{\boldsymbol{\nabla} j^{(k)\,T}\,\mathbf{d} < 0} \qquad \rightarrow \qquad \boxed{-\mathbf{b}_G^{(k)\,T}\,\mathbf{d} < 0}$$

DIC functional's gradient: $\boldsymbol{\nabla} j^{(k)\,T} = -\underbrace{\mathbf{N}^T \boldsymbol{\nabla}\mathbf{G}\left(\mathbf{u}^{(k)}\right)\,\mathbf{r}(\mathbf{x}, \mathbf{u}^{(k)})}_{\approx \mathbf{b}_G^{(k)}}$

$\boldsymbol{\nabla}\mathbf{G}(\mathbf{u})$: a $m \times m$ diagonal matrix that collects the values of $\nabla g \circ \phi(x_p, u)$

How to choose search direction?

$$\left(\text{DD condition: } -{\mathbf{b}_G^{(k)}}^T \mathbf{d} < 0\right)$$

- **Gradient descent** (or Steepest descent):

$$\boxed{\mathbf{d} = \mathbf{b}_G^{(k)}}$$

DD condition OK
Cheap but possibly slow

How to choose search direction?

$\left(\text{DD condition: } -\mathbf{b}_G^{(k)\,T}\,\mathbf{d} < 0\right)$

- **Gradient descent** (or Steepest descent):

$$\boxed{\mathbf{d} = \mathbf{b}_G^{(k)}}$$

DD condition OK
Cheap but possibly slow

- **Newton's method**:

$$\boxed{\mathbf{H}_j^{(k)}\,\mathbf{d} = \mathbf{b}_G^{(k)}}$$

DD condition OK
but costly

$$\text{where} \begin{cases} \text{functional Hessian: } \mathbf{H}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{G}(\mathbf{u}^{(k)}) \nabla \mathbf{G}(\mathbf{u}^{(k)}) \mathbf{N} - \mathbf{N}^T \mathbf{H_g}(\mathbf{u}^{(k)}) \mathbf{R}(\mathbf{u}^{(k)}) \mathbf{N} \\ \text{Image hessian: } \mathbf{H_g}(\mathbf{u}),\ m \times m \text{ diagonal matrix that collect, } H_g \circ \phi(x_p, u) \\ \text{GL Residual: } \mathbf{R}(\mathbf{u}),\ m \times m \text{ diagonal matrix that collect, } r(x_p, u) \end{cases}$$

How to choose search direction?

$\left(\text{DD condition: } -{\mathbf{b}_G^{(k)}}^T \mathbf{d} < 0\right)$

- **Gradient descent** (or Steepest descent):

$$\boxed{\mathbf{d} = \mathbf{b}_G^{(k)}}$$

DD condition OK
Cheap but possibly slow

- **Newton's method**:

$$\boxed{\mathbf{H}_j^{(k)}\,\mathbf{d} = \mathbf{b}_G^{(k)}}$$

DD condition OK
but costly

$$\text{where} \begin{cases} \text{functional Hessian: } \mathbf{H}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{G}(\mathbf{u}^{(k)}) \nabla \mathbf{G}(\mathbf{u}^{(k)}) \mathbf{N} - \mathbf{N}^T \mathbf{H_g}(\mathbf{u}^{(k)}) \mathbf{R}(\mathbf{u}^{(k)}) \mathbf{N} \\ \text{Image hessian: } \mathbf{H_g}(\mathbf{u}),\ m \times m \text{ diagonal matrix that collect, } H_g \circ \phi(x_p, u) \\ \text{GL Residual: } \mathbf{R}(\mathbf{u}),\ m \times m \text{ diagonal matrix that collect, } r(x_p, u) \end{cases}$$

- **Gauss Newton method**:

$$\boxed{\widetilde{\mathbf{H}}_j^{(k)}\,\mathbf{d} = \mathbf{b}_G^{(k)}}$$

DD condition OK
but costly

where the Hessian: $\widetilde{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{G}(\mathbf{u}) \nabla \mathbf{G}(\mathbf{u}) \mathbf{N}$

Fast DIC solvers = Constant RHS Operator

- **Quasi Gauss-Newton**: replace $\boldsymbol{\nabla}\mathbf{G}(\mathbf{u}) \to \boldsymbol{\nabla}\mathbf{F}$

  [Réthoré 10] [Leclerc et al. 11] [Hild & Roux 12] [Passieux & Périé 12] [van Beeck et al. 14] [Wittevrongel et al. 15] [Neggers et al. 16] [Buljac et al. 18]

$$\boxed{\widetilde{\widetilde{\mathbf{H}}}_j \, \mathbf{d} = \mathbf{b}_F^{(k)}}$$

Approx. Hessian: $\widetilde{\widetilde{\mathbf{H}}}_j = \mathbf{N}^T \boldsymbol{\nabla}\mathbf{F} \; \boldsymbol{\nabla}\mathbf{F} \; \mathbf{N}$ (does not depend on (k))

Approx. RHS: $\mathbf{b}_F^{(k)} = \mathbf{N}^T \boldsymbol{\nabla}\mathbf{F} \; \mathbf{r}(\mathbf{u})$

Fast DIC solvers = Constant RHS Operator

- **Quasi Gauss-Newton**: replace $\boldsymbol{\nabla}\mathbf{G}(\mathbf{u}) \to \boldsymbol{\nabla}\mathbf{F}$

  [Réthoré 10] [Leclerc et al. 11] [Hild & Roux 12] [Passieux & Périé 12] [van Beeck et al. 14] [Wittevrongel et al. 15] [Neggers et al. 16] [Buljac et al. 18]

$$\boxed{\widetilde{\widetilde{\mathbf{H}}}_j \, \mathbf{d} = \mathbf{b}_F^{(k)}}$$

Approx. Hessian: $\widetilde{\widetilde{\mathbf{H}}}_j = \mathbf{N}^T \boldsymbol{\nabla}\mathbf{F} \; \boldsymbol{\nabla}\mathbf{F} \; \mathbf{N}$ (does not depend on (k))

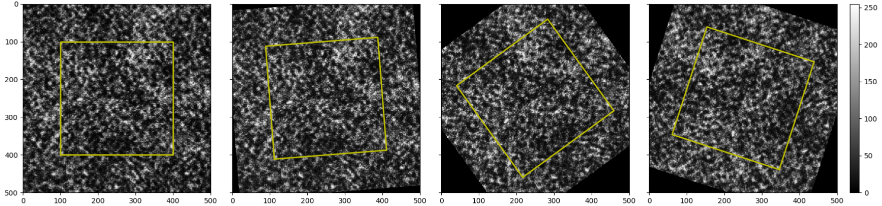
Approx. RHS: $\mathbf{b}_F^{(k)} = \mathbf{N}^T \boldsymbol{\nabla}\mathbf{F} \; \mathbf{r}(\mathbf{u})$

✓ Very fast... but conditionnally fulfills the Descent Direction condition!

Condition: Positivity of $\mathbf{F} = \mathbf{I} + \boldsymbol{\nabla} u$

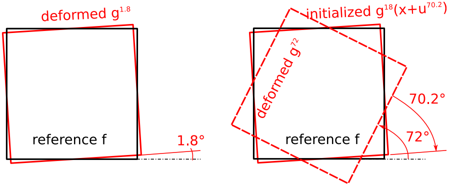
A first example: Rotation

- **Rigid body rotation** $0 < \theta < 180°$ [Neggers et al. 16]



Initialized by $\mathbf{u}^0 = \mathbf{u}(\theta^{ex} - 1.8°)$

Case $\theta = 1.8$

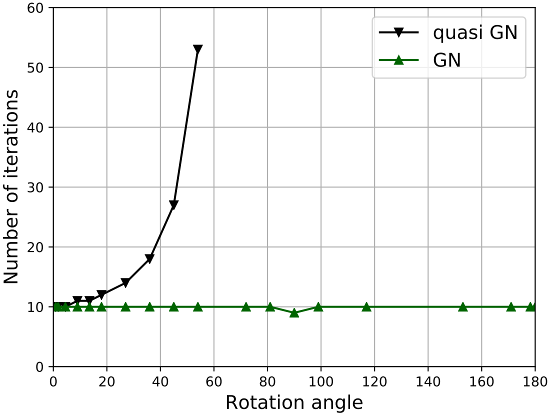


Case $\theta = 72$

with, from a continuous point of view, $g^{1.8}(\mathbf{x}) = g^{72}(\mathbf{x} + \mathbf{u}^{70.2})$

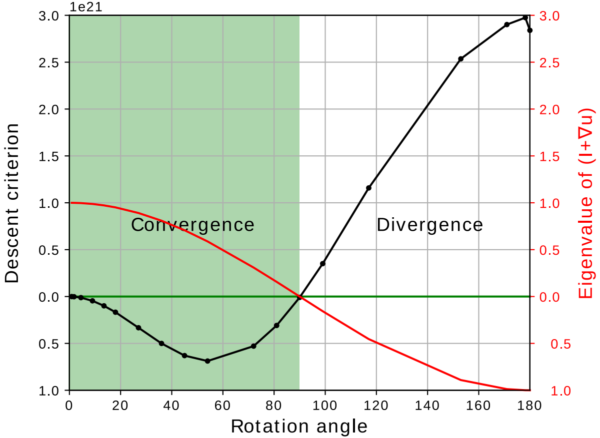
A first example: Rotation

- **Rigid body rotation** $0 < \theta < 180°$ [Neggers et al. 16]



A first example: Rotation

- **Rigid body rotation** $0 < \theta < 180°$ [Neggers et al. 16]



d is a descent direction up to 90°

May require to adjust the step size: $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \alpha \cdot \mathbf{d}$

ICGN FE-DIC

Inverse Compositional Gauss-Newton (ICGN)

- First proposed by **Baker & Matthews** [Baker & Matthews 01]
  Many variants (compositional, inverse, additive, forward...) [Baker & Matthews 04] [Tong 13]
  Subset-DIC [Tong 13] [Pan et al. 13] [Sanchez 16] [Stanier et al. 16]
  Subset-DVC [Pan et al. 14]

Inverse Compositional Gauss-Newton (ICGN)

- First proposed by **Baker & Matthews** [Baker & Matthews 01]
  Many variants (compositional, inverse, additive, forward...) [Baker & Matthews 04] [Tong 13]
  Subset-DIC [Tong 13] [Pan et al. 13] [Sanchez 16] [Stanier et al. 16]
  Subset-DVC [Pan et al. 14]

- **Idea** of classic DIC: find each correction **d** such that

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)} + \mathbf{d}) \qquad \Rightarrow \qquad \widetilde{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{G}(\mathbf{u}^{(k)}) \nabla \mathbf{G}(\mathbf{u}^{(k)}) \mathbf{N}$$

$$\Rightarrow \qquad \overset{\approx}{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{F} \nabla \mathbf{F}\, \mathbf{N}$$

Inverse Compositional Gauss-Newton (ICGN)

- First proposed by **Baker & Matthews** [Baker & Matthews 01]
  Many variants (compositional, inverse, additive, forward...) [Baker & Matthews 04] [Tong 13]
  Subset-DIC [Tong 13] [Pan et al. 13] [Sanchez 16] [Stanier et al. 16]
  Subset-DVC [Pan et al. 14]

- **Idea** of classic DIC: find each correction $\mathbf{d}$ such that

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)} + \mathbf{d}) \qquad \Rightarrow \qquad \widetilde{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{G}(\mathbf{u}^{(k)}) \nabla \mathbf{G}(\mathbf{u}^{(k)}) \mathbf{N}$$

$$\Rightarrow \qquad \widetilde{\widetilde{\mathbf{H}}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{F} \nabla \mathbf{F}\, \mathbf{N}$$

- **Idea** of ICGN: find each correction $-\widetilde{\mathbf{d}}$ such that

$$\mathbf{f} \circ \phi(\mathbf{x}, -\widetilde{\mathbf{d}}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)}) \qquad \Rightarrow \qquad \widetilde{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{F} \nabla \mathbf{F}\, \mathbf{N}$$

Inverse Compositional Gauss-Newton (ICGN)

- First proposed by **Baker & Matthews** [Baker & Matthews 01]
  Many variants (compositional, inverse, additive, forward...) [Baker & Matthews 04] [Tong 13]
  Subset-DIC [Tong 13] [Pan et al. 13] [Sanchez 16] [Stanier et al. 16]
  Subset-DVC [Pan et al. 14]

- **Idea** of classic DIC: find each correction $\mathbf{d}$ such that

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)} + \mathbf{d}) \quad \Rightarrow \quad \widetilde{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{G}(\mathbf{u}^{(k)}) \nabla \mathbf{G}(\mathbf{u}^{(k)}) \mathbf{N}$$

$$\Rightarrow \quad \overset{\approx}{\mathbf{H}}{}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{F} \nabla \mathbf{F} \, \mathbf{N}$$

- **Idea** of ICGN: find each correction $-\widetilde{\mathbf{d}}$ such that

$$\mathbf{f} \circ \phi(\mathbf{x}, -\widetilde{\mathbf{d}}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)}) \quad \Rightarrow \quad \widetilde{\mathbf{H}}_j^{(k)} = \mathbf{N}^T \nabla \mathbf{F} \nabla \mathbf{F} \, \mathbf{N}$$

  ... same operator and right hand side as quasi-GN!!
  The only difference lies in the way the running approximation of $\mathbf{u}^{(k)}$ is updated given $\widetilde{\mathbf{d}}$

ICGN FE-DIC

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

ICGN FE-DIC

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

- with ICGN, **it becomes**:

$$\left(\mathbf{f} \circ \phi(\mathbf{x}, -\widetilde{\mathbf{d}}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)})\right) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})$$

ICGN FE-DIC

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

- with ICGN, **it becomes**:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})$$

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

- with ICGN, **it becomes**:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \underbrace{\phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})}_{\phi(\mathbf{x}, \mathbf{u}^{(k+1)})}$$

ICGN FE-DIC

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

- with ICGN, **it becomes**:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \underbrace{\phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})}_{\phi(\mathbf{x},\mathbf{u}^{(k+1)})}$$

Compute $\mathbf{u}^{(k+1)}$ such that:

$$\phi(\mathbf{x}, \mathbf{u}^{(k+1)}) = \phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})$$

(Inverse Compositional Gauss-Newton)

ICGN FE-DIC

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

- with ICGN, **it becomes**:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \underbrace{\phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})}_{\phi(\mathbf{x}, \mathbf{u}^{(k+1)})}$$

Compute $\mathbf{u}^{(k+1)}$ such that:

$$\phi(\mathbf{x}, \mathbf{u}^{(k+1)}) = \phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})$$

(Inverse Compositional Gauss-Newton)

Optional simplifications:

- first order approximation of the **inversion**: $\phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}}) \approx \phi(\mathbf{x}, \widetilde{\mathbf{d}})$ $\rightarrow$ *forward* method
- zero order approximation of the **displacement**: $\mathbf{u}^{(k))}(\mathbf{x} + \delta) \approx \mathbf{u}^{(k)}(\mathbf{x})$ $\rightarrow$ *additive* method

ICGN FE-DIC

How to **update the running approximation**?

- with (quasi-)GN **it was**:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{d}$$

- with ICGN, **it becomes**:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{g} \circ \underbrace{\phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})}_{\phi(\mathbf{x},\mathbf{u}^{(k+1)})}$$

Compute $\mathbf{u}^{(k+1)}$ such that:

$$\phi(\mathbf{x}, \mathbf{u}^{(k+1)}) = \phi(\mathbf{x}, \mathbf{u}^{(k)}) \circ \phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}})$$

(Inverse Compositional Gauss-Newton)

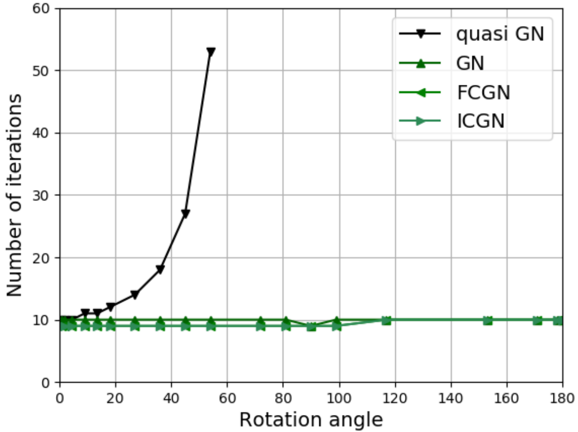
Optional simplifications:

- first order approximation of the **inversion**: $\phi^{-1}(\mathbf{x}, -\widetilde{\mathbf{d}}) \approx \phi(\mathbf{x}, \widetilde{\mathbf{d}})$ $\rightarrow$ *forward* method
- zero order approximation of the **displacement**: $\mathbf{u}^{(k))}(\mathbf{x} + \delta) \approx \mathbf{u}^{(k)}(\mathbf{x})$ $\rightarrow$ *additive* method

Inverse composition unconditionnally fulfills the descent direction criterion.

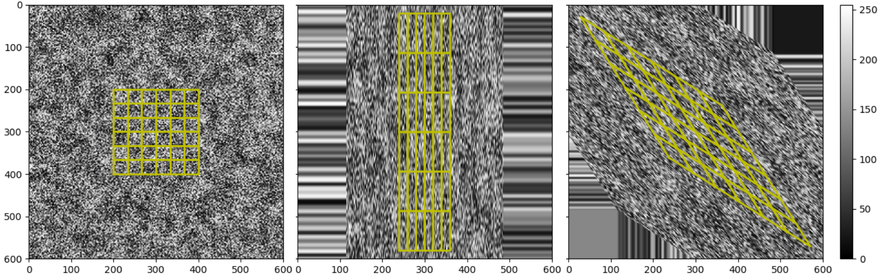
Back to the rotation test case

Number of iterations:



Large deformation test-case #1

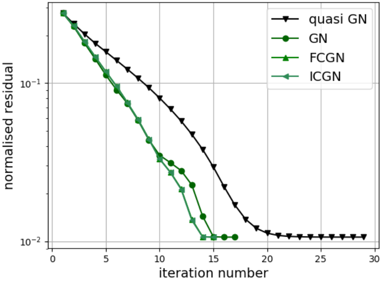
Problem definition

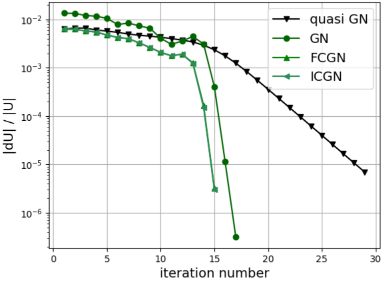


$$\underline{\underline{\varepsilon}}^{normal} = \begin{bmatrix} 1.8 & 0 \\ 0 & -0.4 \end{bmatrix} \quad \text{and} \quad \underline{\underline{\varepsilon}}^{shear} = \begin{bmatrix} 0.7 & -1.1 \\ -1.1 & 0.7 \end{bmatrix}$$

Large deformation test-case #1

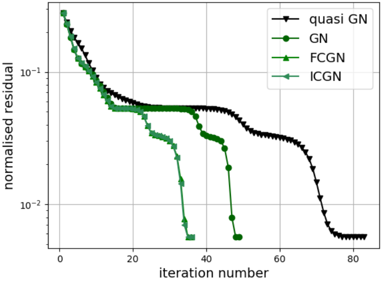
Convergence of the tension test-case:

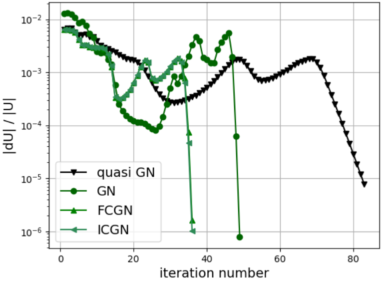




Large deformation test-case #1

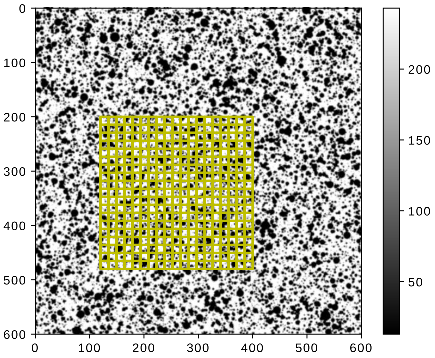
Convergence of the shear test-case:





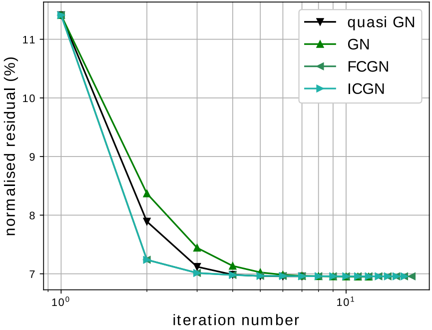
Large deformation test-case #2

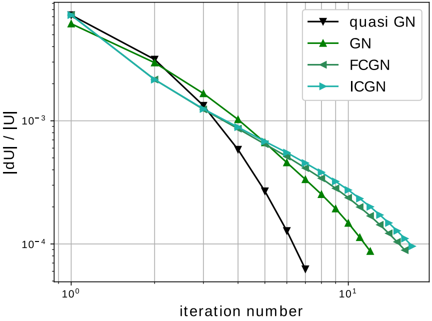
Punch image set (Courtesy of **K. Genovese**) (generated using BSpeckleRender [Sur et al. 18])



Large deformation test-case #2

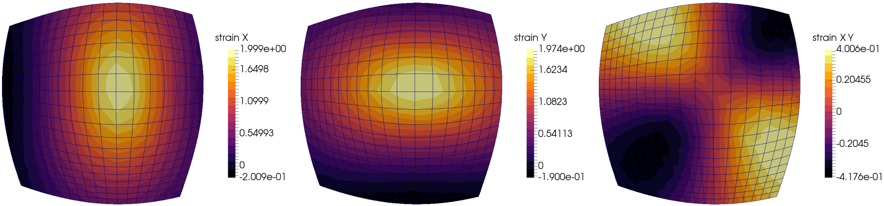
Convergence:





Large deformation test-case #2

Heterogeneous strain field



SOLVER: Summary

Descent-based DIC solvers vs **finite strain**

- numerically efficient because we have a **semi-analytic expression of the gradient**.
- Steepest Descent, Newton, Gauss-Newton, quasi-Gauss Newton and Inverse Compositional Gauss-Newton **should work in general**.
- if **large rotations** to be expected ($> 20^\circ$), the classic approx. $\boldsymbol{\nabla}\mathbf{F} \approx \boldsymbol{\nabla}\mathbf{G}(\mathbf{u}^{(k)})$ may be slow or may require line search (to adapt the step size α)

  $\rightarrow$ Prefer **Inverse Compositional Gauss Newton** (ICGN)

2. INITIALIZATION

Initialization

All the above algorithms require to initialize close to the (unknown) solution.
Different possibilities (that may be combined):

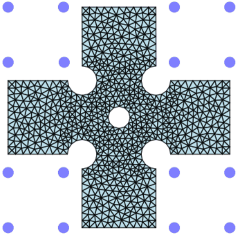
Initialization

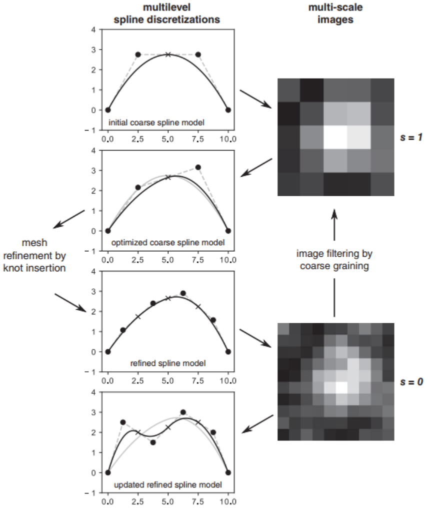
- **Multilevel Initialization:**
  image coarsening or filtering
  kinematic model coarsening.

Increasing mesh size + projection [Rethore et al. 08]
Weak regularization [Passieux et al. 12]
Strong regularization [Rouwane et al. 23]

$$\boldsymbol{\alpha} = \operatorname{argmin} \frac{1}{2} \|\mathbf{f} - \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{R}\ \boldsymbol{\alpha})\|^2$$

FFD [Chapelier et al. 21]





Initialization

- **Multilevel Initializations**
- **Incremental approaches**, if a sufficient number of time increment is available in the image series.
  For step i initialized with solution of previous step $\mathbf{u}^{(i-1)}$:

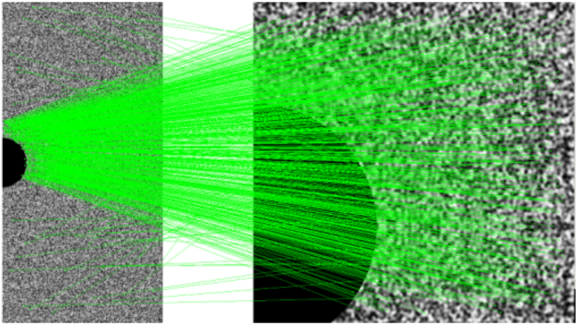
$$\mathbf{d} = \mathrm{argmin} \frac{1}{2} \ \|\mathbf{f}_0 - \mathbf{f}_i \circ \phi(\mathbf{x}, \mathbf{u}^{(i-1)} + \mathbf{d})\|^2$$

  Or Incremental approach (updated reference image):

$$\mathbf{d} = \mathrm{argmin} \frac{1}{2} \ \|\mathbf{f}_{i-1} - \mathbf{f}_i \circ \phi(\mathbf{x}, \mathbf{d})\|^2$$

Initialization

- **Multilevel Initializations**
- **Incremental approaches**
- **Point matching algorithms** SIFT [Lowe 04] SURF [Bay et al. 08]
  DIC: [Zhou et al. 12] [David et al. 14] [Genovese et al. 18]
  + possible outliers cleaning / regularization procedures



Initialization

- **Multilevel Initializations**
- **Incremental approaches**
- **Point matching algorithms**
- **Reliability/Quality-guided** subset approaches [Pan 09, Zhou et al. 12]

- No universal tool, maybe case dependent.
- **Small steps:** combination of incremental approach and strong regularization usually solves the problem.
- **Large steps:** point matching algorithms are efficient because scale invariants.

3. REGULARIZATION

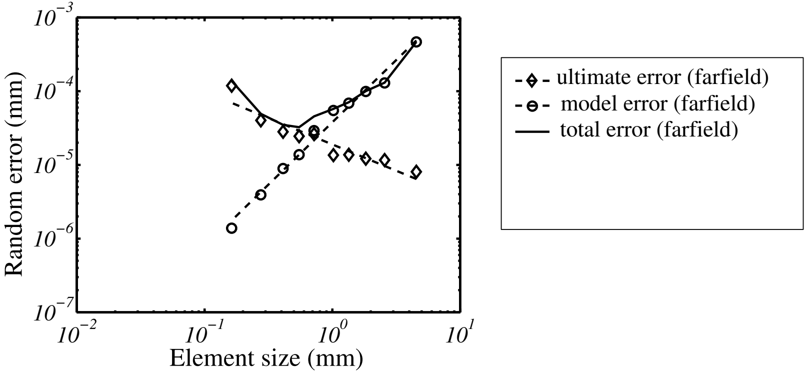
Graylevel conservation problem is ill-posed:
DIC requires regularisation

Regularization

Adjusting the subset/element size is **a strong regularisation technique**

Regularization

Adjusting the subset/element size is **a strong regularisation technique**

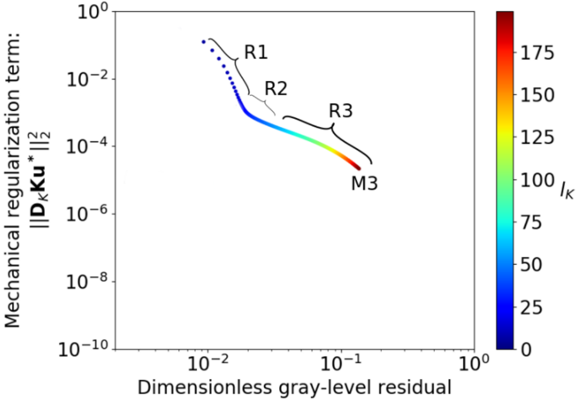


... but it's not the unique (not the best?) way.

Regularization

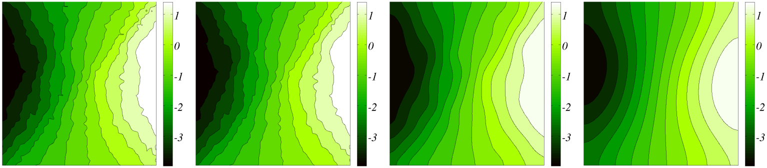
- An alternative: **Weak regularization** (Tikhonov)
  [Réthoré, Roux, Hild 08, Leclerc et al. 10, JCP & Périé 12, Rouwane et al. 22]
  α is a **filter cutoff** length - can be set L-curve [Hansen 00] :

$$\mathbf{d} = \text{argmin} \underbrace{\frac{1}{2} \|\mathbf{f} - \mathbf{g} \circ \phi(\mathbf{x}, \mathbf{d})\|^2}_{\textit{graylevelterm}} + \alpha \underbrace{\frac{1}{2} \|\mathbf{d}\|^2_{\mathbf{A}}}_{\text{regularization term}}$$



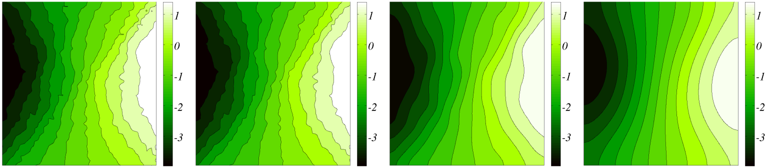
Regularization

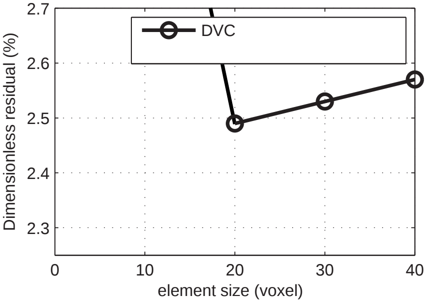
- Example of Laplacian regularization [Perini et al. 14] :



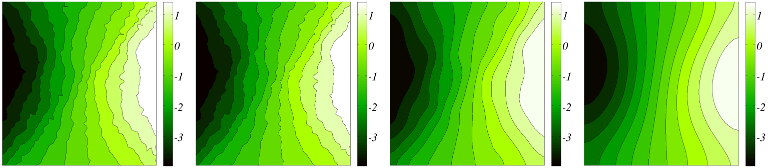
Regularization

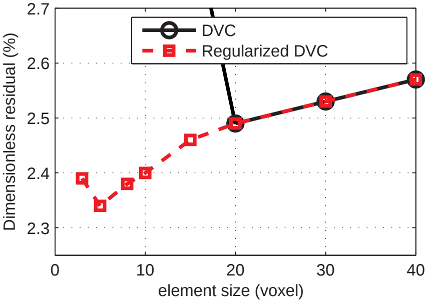
- Example of Laplacian regularization [Perini et al. 14] :





- Example of Laplacian regularization [Perini et al. 14] :





Regularization

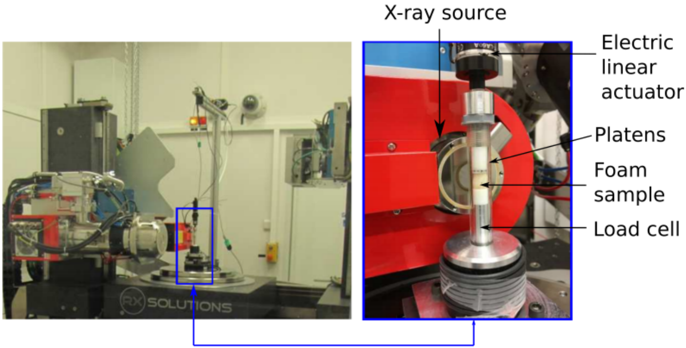
Many choices for **A**

- **Rigid body** [Staring et al. 07] [JCP et al. 14]
- **Laplacian** [Rohlfing et al. 03] [JCP and Périé 12]
- **Elastic** [Bajcsy and Kovacic 89] [Kybic and Unser 03] [Likar and Pernus 01] [Réthoré et al. 09] [Leclerc et al. 11] [Mendoza et al. 19] [Rouwane et al. 22, 23]
- **Elasto-plastic** [Réthoré et al. 13] [Mathieu et al. 15]
- **Finite strain** [Genet et al. 18]

[Bajcsy and Kovacic 89] Multiresolution elastic matching, Comput. Vis. Graphics Image Process., 46(1)1–2.
[Kybic and Unser 03] Fast parametric elastic image registration, IEEE Trans. Image Process., 12(11)1427–1442
[Likar and Pernus 01] A hierarchical approach to elastic registration based on mutual information, Im. Vis. Comput.
[Staring et al. 07] A rigidity penalty term for nonrigid registration, Med. Phys., 34(11)4098–4108
[Rohlfing et al. 03] Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint, IEE Trans. Med. Imag.
[Réthoré et al. 09] An extended and integrated digital image correlation technique applied to the analysis of fractured samples, Eur. J. Comp. Mech.
[Leclerc et al. 11] Voxel-scale digital volume correlation. Experimental Mechanics.
[JCP and Périé 12] High resolution digital image correlation using Proper Generalized Decomposition: PGD-DIC, Int. J. Numer. Meth. Engrg.
[Réthoré et al. 13] Robust identification of elasto-plastic constitutive law parameters from digital images using 3D kinematics, Int. J. Sol. Str.
[JCP et al. 14] A digital image correlation method for tracking planar motions of rigid spheres: application to medium velocity impacts. Exp. Mech.
[Mathieu et al. 15] Estimation of Elastoplastic Parameters via Weighted FEMU and Integrated-DIC. Exp. Mech.
[Genet et al. 18] Equilibrated warping: Finite element image registration with finite strain equilibrium gap regularization. Medical image analysis 50, 1-22.

Mechanically regularized DVC with large local bendings

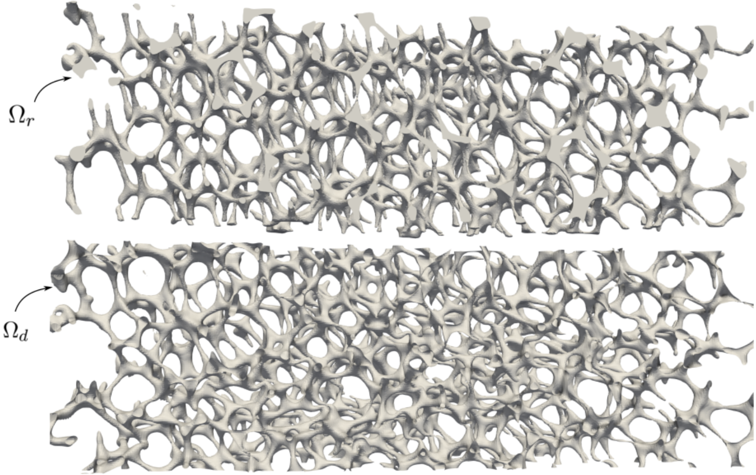
In situ compression test on a open-cell foam [Pétureau 18] .



No pattern at the strut scale!

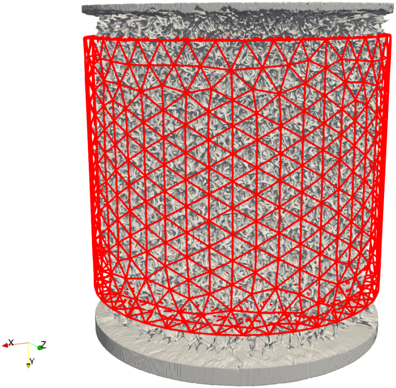
Mechanically regularized DVC with large local bendings

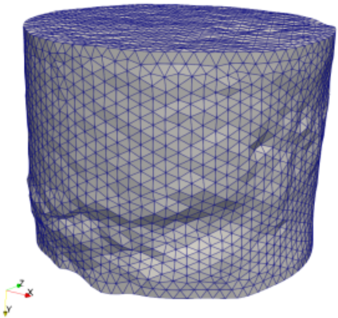
A double challenge: **No pattern at the strut scale** AND **Large local buckling/bending**



Mechanically regularized DVC with large local bendings

Classic (FE) DVC at an *homogenized* scale : **element size** $>$ **cell size**



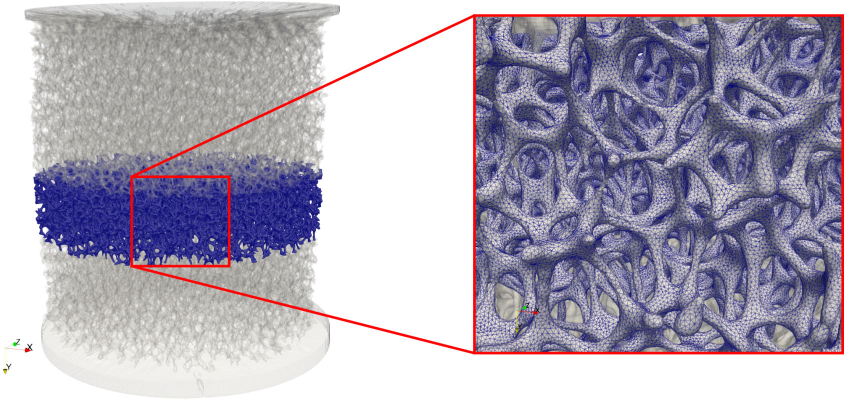


Even in this situation, it requires a bit of skills (the shape of the pattern dots evolves):

- **multiscale** initialisation + **reduced kinematic basis**
- image **filtering**
- **adjusted weak regularisation** based of the Laplacian of the displacement field

Mechanically regularized DVC with large local bendings

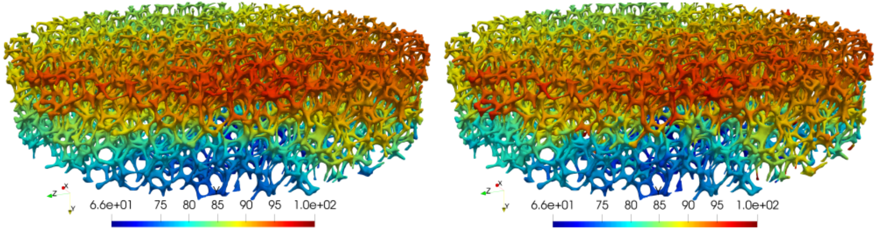
Proposed architecture-driven DVC: [Rouwane et al. Exp. Mech. 2023]
An **image-based geometric and mechanical model** used to weakly regularize DVC.



- Voxel based: thresholding [Hollister et al. 94] , graylevel values [Liu et al. 19]
- **Fitted mesh**: Marching cubes [Lorensen et al. 87] [Frey et al. 94]
- Unfitted mesh: FCM, X-FEM, CUT-FEM, Ficticious domain method [Schilinger et al. 11,12,15] [Burman et al. 15] [Verhoosel et al. 15] [Lehrenfeld et al. 16] [Fries et al. 16] [Legrain et al. 18] [Kerfriden et al. 20]

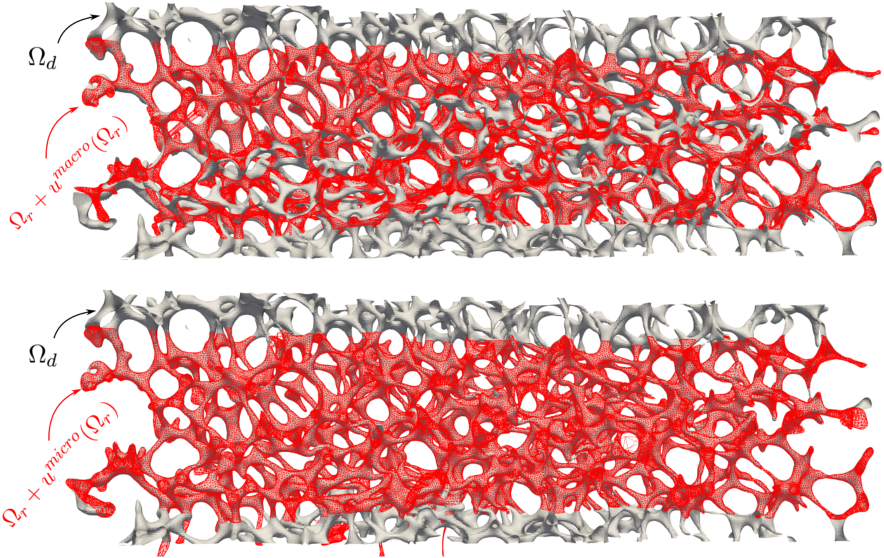
Mechanically regularized DVC with large local bendings

Macro: coarse mesh + FE-DVC + Tikhonov regularisation
Micro: Proposed architecture-driven DVC assisted by an image-based model
Visually comparable **displacement fields** on top of the fitted (micro) mesh



(a) Macro solution. (b) Micro solution.

Mechanically regularized DVC with large local bendings

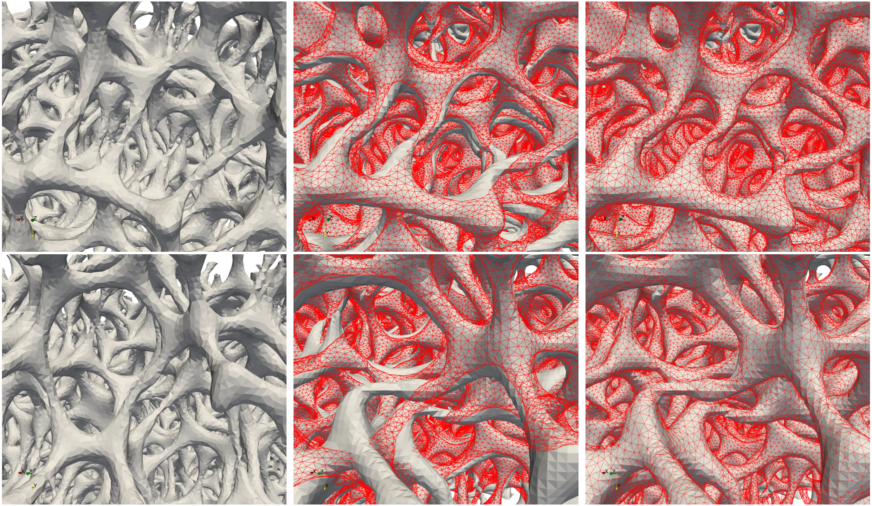


Mechanically regularized DVC with large local bendings

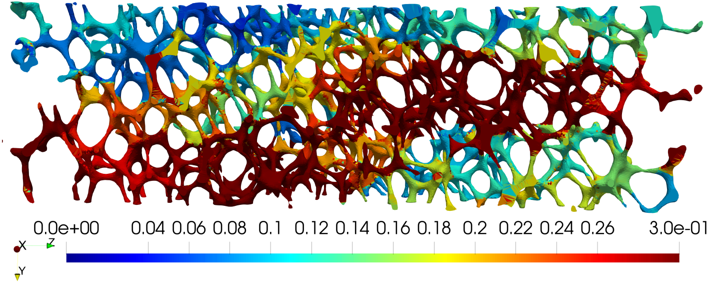
Reference Image

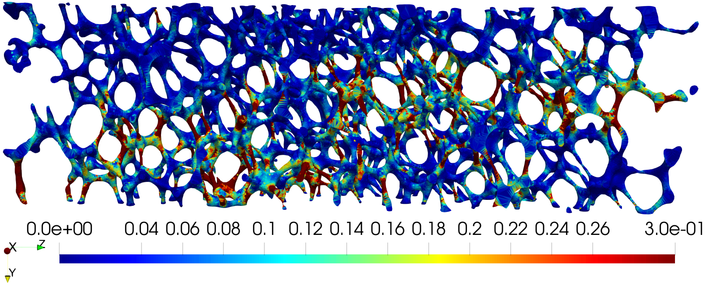
Deformed Image
+ macro displacement

Deformed Image
+ micro displacement



Mechanically regularized DVC with large local bendings





REGULARIZATION: Summary

- Digital Image Correlation **must be regularized**
- **Adjusting the subset/element size** is a **strong regularisation** technique and is not always the best option
- **Weak regularisation** is a very interesting alternative especially in **finite strain/rotations** where subset/elements deforms a lot
- **Weak linear elastic regularisation** works well to help DVC **outside small perturbations**

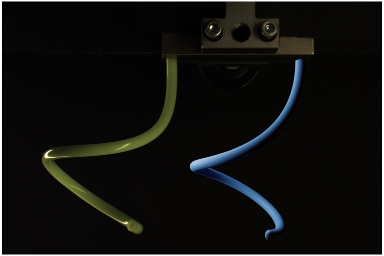
4. OCCULTATION, LIGHT EFFECTS and PIXEL SIZE

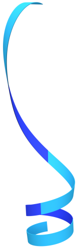
$\rightarrow$ Toward a new photometric formulation

Toward Photometric DIC

How to perform DIC in such nonlinear problems?

- Tape spring hinges [Soykasap 07]
- Nonlinear slender elastic structures [Romero et al. 21] [Charrondière et al. 21]





Many problems:

- Occultation
- Surfaces not visible in both reference and deformed configurations
- Lighting effects
- Large displacements/rotations

Toward Photometric DIC

Classic DIC is formulated on the comparison between two images:

$$f(\mathbf{x}) - g(\mathbf{x} + \mathbf{u}(\mathbf{x}))$$

Proposed Approach: comparison of a model and a measurement [Fouque et al. 21, 22]

$$g(\mathbf{x}) - \hat{I}(\mathbf{x}, \mathbf{u})$$

where $\hat{I}$ is a **synthetic image** build from a **model**

Toward Photometric DIC

Classic DIC is formulated on the comparison between two images:

$$f(\mathbf{x}) - g(\mathbf{x} + \mathbf{u}(\mathbf{x}))$$

Proposed Approach: comparison of a model and a measurement [Fouque et al. 21, 22]

$$g(\mathbf{x}) - \hat{I}(\mathbf{x}, \mathbf{u})$$

where $\hat{I}$ is a **synthetic image** build from a **model**

We need:

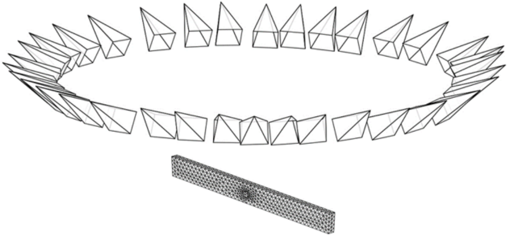
- to know the pattern [Shi et al. 23] or to learn it from a scan [Fouque et al. 22]
- to learn and synthetise the generation of a pixel
- to learn and synthetise the way light interacts with the specimen
- to model and learn the lighting conditions

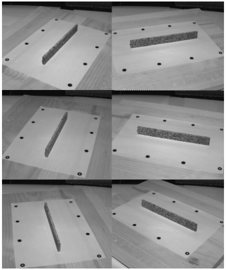
Inspiration from the CV community [Birkbeck et al. 16, Goldlücke et al. 14]

Toward Photometric DIC

Step 1: Building the digital twin from a multiview scan of the object

- Only **one camera** used
- **Multi-view** pictures instead of several cameras ≈ **scan**





- Regularisation strategy: **Amount of available data increase**

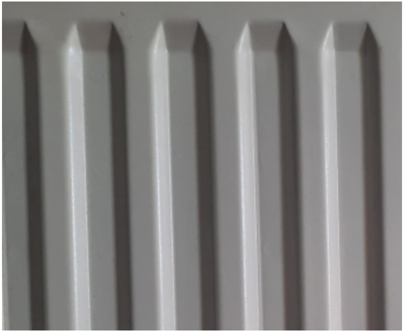
Toward Photometric DIC

Step 1: Building the digital twin

- **Key idea** = Retrieve the *intrinsic texture* [Dufour et al. 15] of the specimen from all the observations
- **Need** = Rendering equation mapping an intrinsic texture to greylevel under given lighting conditions :

Assuming a Lambertian model, we have:

$$\boxed{R_i(\mathbf{x}) = \langle \mathbf{n}(\mathbf{x}), \mathbf{l}_i \rangle \; I_i \; T(\mathbf{x})}$$



- R_i: graylevel rendered of picture i
- $\mathbf{n}$: outgoing unit normal vector to the specimen surface at point $\mathbf{x}$
- $\mathbf{l}_i$: Unit vector pointing from $\mathbf{x}$ in the direction of the light
- I_i: light intensity
- T: the sought intrinsic texture: **albedo**

Toward Photometric DIC

- Residual based on:

$$\underbrace{R_i(\mathbf{x})}_{\text{predicted}} \; - \underbrace{f_i \circ \mathbf{P}_i(\mathbf{x})}_{\text{observed}}$$

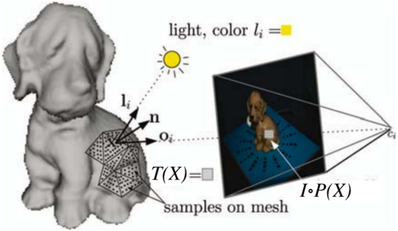
- Minimisation of the functional:

$$F(T, \mathbf{S}, \mathbf{p}^i_{ext}) = \sum_i \int_\Omega V'_i(\mathbf{x}) \Big[\langle \mathbf{n}(\mathbf{x}), \mathbf{l}_i \rangle \; l_i \; T(\mathbf{x}) - f_i \circ \mathbf{P}_i(\mathbf{x} + \mathbf{S}(\mathbf{x})) \Big]^2$$

where V'_i is a weighting term:

$$V'_i(\mathbf{x}) = V_i(\mathbf{x}) \;\; \langle \mathbf{n}(\mathbf{x})), \mathbf{O}_i(\mathbf{x}) \rangle$$

V_i is the visibility function (0 or 1)
the second terms account for the fact that a more reliable sampling of the texture is achieved by nearly fronto parallel observations.



[Birkbeck et al. 16]

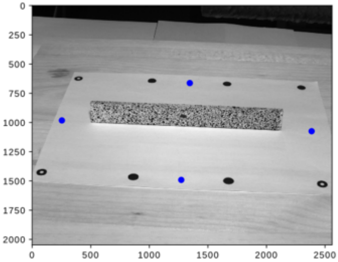
Toward Photometric DIC

Texture T, Shape $\mathbf{S}$ and Extrinsic parameters $\mathbf{p}_{ext}$ minimise functional:

$$T, \mathbf{S}, \mathbf{p}_{ext}^{i} = \operatorname{argmin} \sum_{i} \int_{\Omega} V_i'(\mathbf{x}) \left[\langle \mathbf{n}(\mathbf{x}), \mathbf{l}_i \rangle \ l_i \ T(\mathbf{x}) - f_i \circ \mathbf{P}_i(\mathbf{x} + \mathbf{S}(\mathbf{x})) \right]^2$$

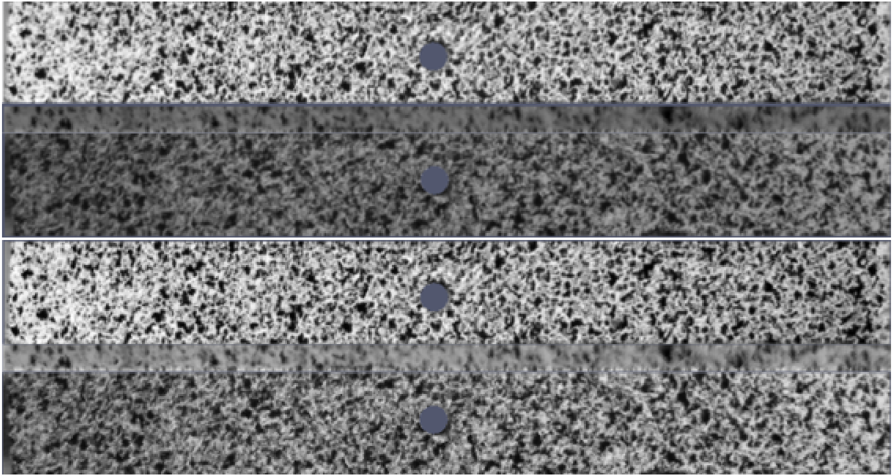
Additional assumptions:

- light source behind the camera: $\forall i,\ \mathbf{O}_i = \mathbf{Z}_i$
- light must be calibrated. 4 points on the white paper T=1.
- minimisation are performed alternatively.



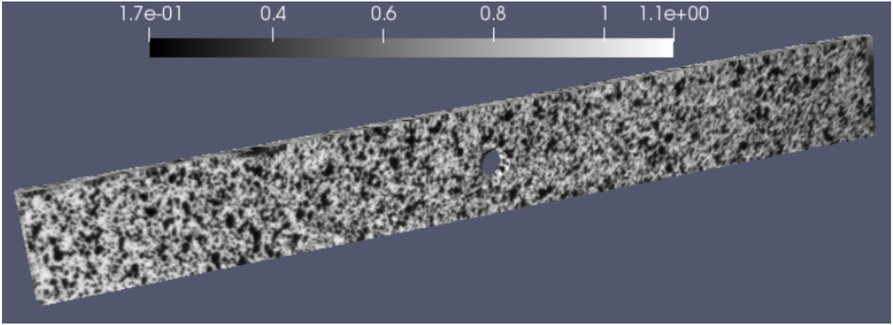
Toward Photometric DIC

Results: (top) graylevel and (bottom) albedo



Toward Photometric DIC

Reconstruction of a textured digital twin of the sample:

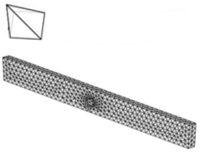


Toward Photometric DIC

Step 2: Displacement measurement using the digital twin
Minimisation of the same functionnal:

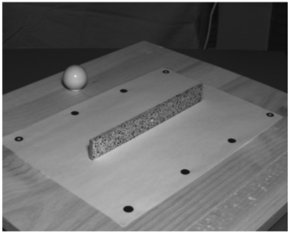
$$\mathbf{U} = \operatorname{argmin} \sum_i \int_\Omega V_i'(\mathbf{x}) \Big[\langle \mathbf{n}(\mathbf{x}), \mathbf{l}_i \rangle \; l_i \; T(\mathbf{x}) - f_i \circ \mathbf{P}_i(\mathbf{x} + \mathbf{U}(\mathbf{x})) \Big]^2$$

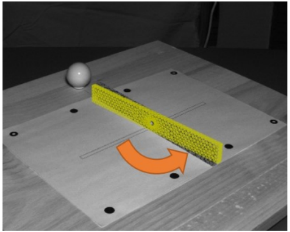
But this time T and $\mathbf{p}_{ext}$ are fixed and (here) one single camera:

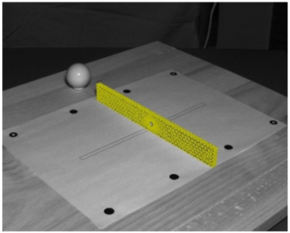


Toward Photometric DIC

Step 2: Displacement measurement using the digital twin



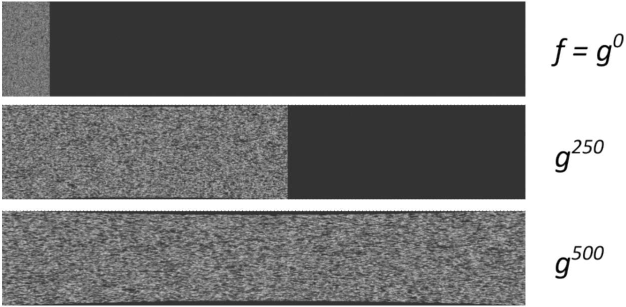




The algorithm, initialization close to the solution (95°) is able to estimate a rigid body rotation of 90° even if the visible surfaces are different in the reference and moved configuration seen by the same camera.

Toward Photometric DIC

To be done: Generation of a pixel
Example of a very large deformation synthetic dataset (BSpeckleRender [Sur et al. 18])

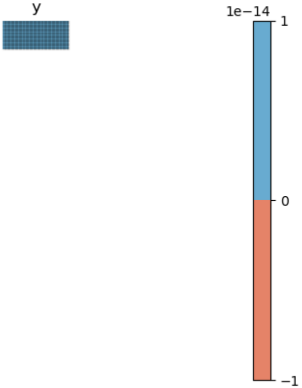


[Courtesy of B. Blaysat and M. Coret]

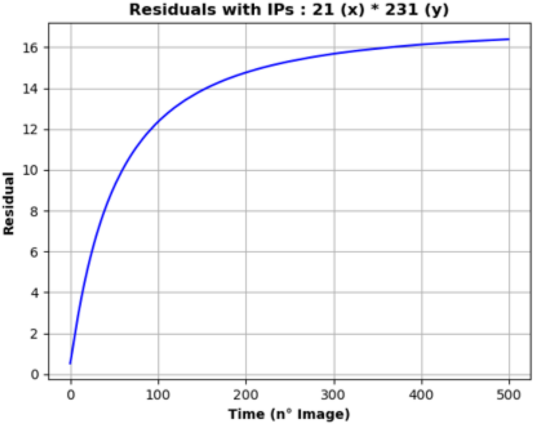
A series of **500 images** up to **700% of strain**!

Toward Photometric DIC

No problem to make it converge with an incremental approach:



Toward Photometric DIC



Conclusions

Many things already working

- Solvers work, if large rotations → ICGN
- Many initialization solutions
- Even elastic regularisation works in large displacements/strain

Conclusions

Many things already working

- Solvers work, if large rotations → ICGN
- Many initialization solutions
- Even elastic regularisation works in large displacements/strain

But a lot to be done

- Ph-DIC to be developed and tested in the presence of specimen deformation
- Ph-DIC capable to take into account the integration of light in the pixel

Conclusions

Many things already working

- Solvers work, if large rotations $\rightarrow$ ICGN
- Many initialization solutions
- Even elastic regularisation works in large displacements/strain

But a lot to be done

- Ph-DIC to be developed and tested in the presence of specimen deformation
- Ph-DIC capable to take into account the integration of light in the pixel

pyxel

谢谢