A SAEM TECHNIQUE FOR FINITE-ELEMENT-BASED REGISTRATION OF IMAGE SEQUENCES

FRÉDÉRIC RICHARD AND ADELINE SAMSON

Abstract.

1. Introduction

2. Registration Model

2.1. **A scene model.** Through a series of images, we observe a non-rigid object which has undergone some deformations in a time interval $[0, T]$. In our applications, the observed object is either an organ or a part of the human body. For describing deformations of the object, we use a common reference domain denoted $\mathcal{D}$, which is a connected subset of $\mathbb{R}^3$. As shown on Figure 1, we define a mapping ϕ of $\mathcal{D} \times [0, T]$ onto $\mathbb{R}^3$ which gives the position $\phi(z, t)$ at any time t of a material point which is located at any point z in the reference domain. Each map $\phi(\cdot, t)$ is one-to-one and has an inverse denoted $\psi(\cdot, t)$; the value $\psi(x, t)$ gives the position in the reference domain of a material point which is located in x at time t. In continuum media mechanics, coordinates z of the reference domain are called Lagrangian whereas coordinates x in the deformed domain are called Eulerian. Physical deformations of the object are unknown.

Date: August 6, 2007.

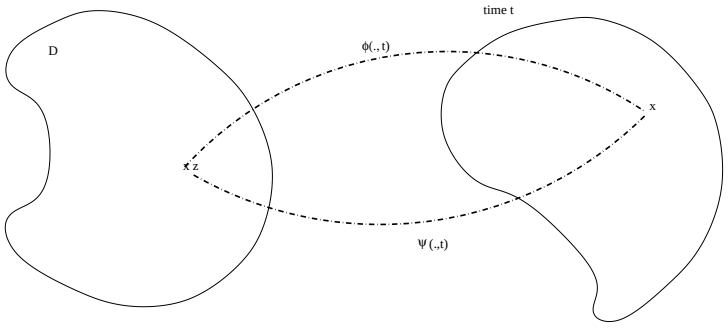

Figure 1. example caption

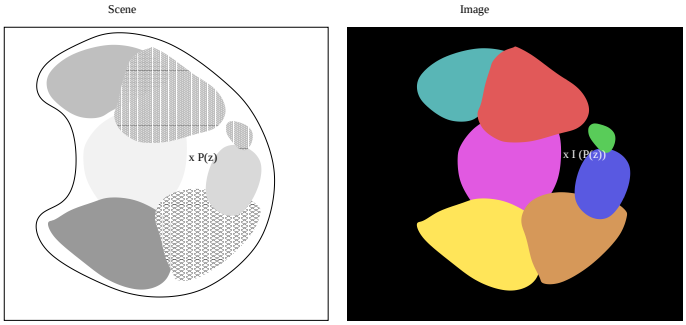

FIGURE 2. example caption

As illustrated on Figure 2, the object observed throughout the image sequence is made of several tissues which have different material properties and is thus non-homogeneous. We define on $\mathcal{D}$ an application $\mathcal{M}$ which gives a description of the material properties of the object in the reference configuration. This application is unknown. It is introduced here for the sake of the presentation, but as we will see later, it will not be used as such.

In this paper, we will only present a 2D implementation of our method. The observations will exclusively concern a part of the object which is located in an horizontal plane of $\mathbb{R}^3$. We will assume that while it deforms, the part of the object in the plane remains within the plane, ie, at points of the plane, there is no displacement of the objects in the vertical direction. In the remaining of the text, $\mathcal{D}$ will refer to a connected set in the plane of interest. Points of $\mathcal{D}$ will be located by their two coordinates on the plane. The application ϕ will be identified with its restriction to the plane and have values in $\mathbb{R}^2$.

2.2. **Observation model.** As illustrated on Figure 2, we assume that gray-level values observed in images are related to the physical properties of the object. Due to an acquisition noise, the image is however an imperfect observation of the physical properties of the object. We denote by f the noise-free image of the object as positioned in the reference domain. Due to the above assumption, f is of the form

$$f = \mathcal{I}(\mathcal{M}), \tag{1}$$

where $\mathcal{I}$ represents the action of the image acquisition system which transforms the object properties into a signal. Here, we could use a physical acquisition model for defining precisely the action $\mathcal{I}$. In this paper, we will however work exclusively on the image f and not on the physical properties $\mathcal{M}$ the images are related to. So we do not need a precise definition of $\mathcal{I}$. Mathematically, the image f is defined as a mapping of $\mathcal{D}$ into $\mathbb{R}$. It will be called the template.

In the image sequence, the object is observed as it deforms from a fixed view point. At any fixed observation point x, we observe a series g of signals

$$\forall\, t \in [0,T], g(x,t) = \mathcal{I}(\mathcal{M} \circ \psi(\cdot,t))(x) + b(x,t), \tag{2}$$

where b is an additive noise and $\mathcal{I}(\mathcal{M} \circ \psi(\cdot,t))(\cdot)$ is the noise-free image of the deformed object at time t; $\mathcal{I}(\mathcal{M} \circ \psi(\cdot,t))(x)$ reflects the properties of the material points which pass through the position x at time t. We further assume that the noise-free image of the deformed object is equal to the deformed image of the object observed in the reference domain, leading to the expression

$$\mathcal{I}(\mathcal{M} \circ \psi(\cdot,t))(x) = \mathcal{I}(\mathcal{M})(\psi(x,t)) = f(\psi(x,t)). \tag{3}$$

We then have the following observation model.

$$\forall\, x, \forall\, t \in [0,T], g(x,t) = f(\psi(x,t)) + b(x,t). \tag{4}$$

This observation model is defined on a continuum. In practice, images are however observed on a fixed and finite set $\mathcal{G}$ of points of $\mathbb{R}^3$, called the image grid:

$$\mathcal{G} = \{x_n, n = 1, \cdots, n_g\}. \tag{5}$$

The image are also acquired on a limited set of n_t times $\{t_1, \cdots, t_{n_t}\}$.

Let us denote by Y_{in} the gray level value $g(x_n, t_i)$ which is observed at position x_n and at time t_i and by B_{in} the random noise at same position and time. The discrete observation model can then be defined as

$$\forall i \in \{1, \cdots, n_t\}, \forall\, n \in \{1, \cdots, n_g\}, Y_{in} = f(\psi(x_n, t_i)) + B_{in}. \tag{6}$$

2.3. **Template model.** We assume that we can partition the domain $\mathcal{D}$ into regions where the material properties of the object are homogeneous ($\mathcal{M}$ is constant). We will denote by p_e the regions of the partition and by $\dot{p}_e$ the interiors of these regions. We have

$$\mathcal{D} = \cup_{e=1}^{n_e} p_e \quad \text{and} \quad \forall e_1 \neq e_2, \dot{p}_{e_1} \cap \dot{p}_{e_2} = \emptyset.$$

Due to the above assumptions, the template f is constant on each element of the partition. Let ρ_e be the gray-level value of f on the region p_e. The template can then be written as

$$\forall\, z \in \mathcal{D}, f(z) = \sum_{e=1}^{n_e} \rho_e \mathbb{1}_{z \in p_e}. \tag{7}$$

In practice, the partition of $\mathcal{D}$ is defined by triangulation. The regions p_e are triangle of the plane which are called finite elements and defined by three vertices. We will denote by $\mathcal{N}$, the set of all the vertices of the triangulation, by n_v the number of vertices, and by $\mathcal{N}_e$ the set of vertices defining the finite element p_e. The coordinates of the jth vertex in $\mathcal{D}$ will be denoted by $\nu_j = (\nu_{j1}, \nu_{j2}) \in \mathbb{R}^2$. Note that the template model we described can be easily extended to 3D by partitioning the domain $\mathcal{D}$ with tri-dimensional simplices.

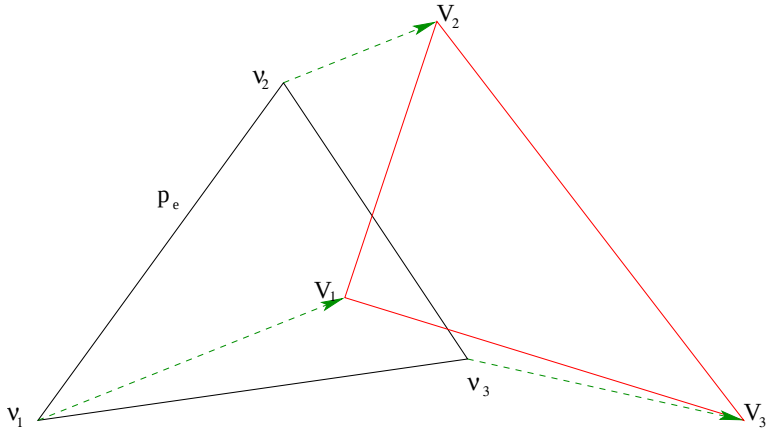

FIGURE 3. example caption

2.4. **Deformation model.** We describe deformations $\phi(\cdot, t_i)$ with piecewise-affine functions: let $\phi(\cdot, t_i) = z + u(\cdot, t_i)$, where id is the identity map and u are displacements. We assume that for all time t_i, displacements $u(\cdot, t_i)$ are affine on each finite elements. Deformations $\phi(\cdot, t_i)$ are thus continuous and invertible on $\mathcal{D}$. Furthermore, the image $\phi(p_e, t_i)$ of a finite element p_e by $\phi(\cdot, t_i)$ remains a triangle (denoted by p_{ei}). As shown on Figure 3, the vertices of the triangle p_{ei} are the images of the vertices of p_e. Let V_{ij} denote the position at time t_i of the jth vertex of the triangulation. The triangle p_{ei} is entirely defined by the set of vertex positions $\{V_{ij}, j \in \mathcal{N}_e\}$, which is written formally as $p_{ei} = \mathcal{P}(V_{ij}, j \in \mathcal{N}_e)$. Moreover, deformations $\phi(\cdot, t_i)$ at time t_i are completely characterized by the set of vertex positions $\{V_{ij}, j = 1, \cdots, n_v\}$.

Within the piecewise-affine framework, we introduce a stochastic representation of deformations. At each time t_i, we represent the vector $\mathbf{V}_i = (V_{ij})_{j=1}^{n_v}$ of vertex positions as a random vector. The definition of the distribution law governing these vectors strongly depends on the application at hand. In our application, we can reasonably assume that vectors $\mathbf{V}_i$ are independant and identically distributed. We represent vectors $\mathbf{V}_i$ as variations around positions $\nu = (\nu_j)_{j=1,\cdots,n_v}$ of the vertices in the reference domain. We define the law governing vectors $\mathbf{V}_i$ as multidimensional Gaussian distribution with mean ν and covariance matrix Σ. Note that in some applications, the assumption that vectors $\mathbf{V}_i$ are independant may not be realistic. In such a situation, we can extend our approach by describing distributions of $\mathbf{V}_i$ given $\mathbf{V}_{i-1}, \cdots, \mathbf{V}_1$.

The covariance matrix Σ is formed of elements which are model parameters to be estimated. The number of different elements is equal to $2((n_v-1)^2+n_v)$, where n_v is the number of vertices. In order to reduce the number of model parameters, we made some additional assumptions on vertex displacement dependancies. For that, we define a neighborhood structure on the vertex set $\mathcal{N}$. We say that two vertices are neighbors if and only if they are distinct

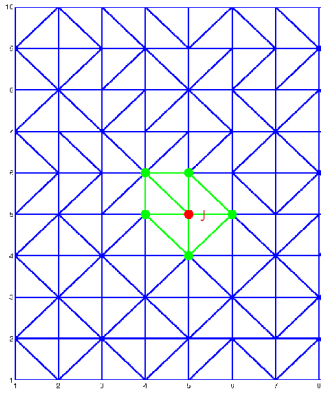

FIGURE 4. example caption

and if they belong to a same finite element (see Figure 4 for an illustration). Equipped with such a neighborhood structure, the vertex set $\mathcal{N}$ is a graph. On the graph $\mathcal{N}$, we then assume that displacements of two different non-neighbor vertices $j \neq k$ are independant ie,

$$\mathbb{E}((V_{ij} - \nu_j)(V_{ik} - \nu_k)^t) = (0). \tag{8}$$

In other words, we only take into account local dependancies between vertex displacements. Consequently, the covariance matrix Σ is sparse. *****just un petit problem: l'inverse est normalement sparse de la meme facon, non?***

2.5. **The complete statistical model.** Due to the template and deformation representations introduced above, we can simplify the expression of the term $f(\psi(x_n, t_i))$ in Equation (6). Replacing f by its expression given by Equation (7), we first obtain

$$f(\psi(x_n, t_i)) = \sum_{e=1}^{n_e} \rho_e \mathbb{1}_{\psi(x_n,t_i)\in p_e}.$$

But, the point $\psi(x_n, t_i)$ is in p_e if and only if the point x_n is in $p_{ei} = \phi(p_e, t_i)$. Hence, we can write

$$f(\psi(x_n, t_i)) = \sum_{e=1}^{n_e} \rho_e \mathbb{1}_{x_n\in\mathcal{P}(V_{ij},j\in\mathcal{N}_e)}.$$

In this expression, the inverse deformation ψ is not present anymore. Deformations are described in a Lagrangian way using only variables V_{ij}. Such a property is important from a practical point of view. It enables to do the computations without having to inverse ϕ.

Taking into account the simplification of $f(\psi(x_n, t_i))$, we can now define the final observation model:

$$\forall i \in \{1, \cdots, n_t\}, \forall\, n \in \{1, \cdots, n_g\}, Y_{in} = \sum_{e=1}^{n_e} \rho_e \mathbb{1}_{x_n \in \mathcal{P}(V_{ij}, j \in \mathcal{N}_e)} + B_{in}. \tag{9}$$

Let us recall what are the different components of this model. Points x_n of the grid and times t_i are fixed deterministic variables. Image gray-level values Y_{in} at positions x_n and time t_i are the observations. Vectors V_{ij} are positions of the vertex j at time i in the finite-element decomposition of the domain. These vectors are random and unobserved. Variables B_{in} represent a random additive noise. Finally, the model has a set of different parameters:

- the template gray-level values $\rho = (\rho_e)_{e=1}^{n_e}$,
- the variance σ^2 of the additive noise (we assume that the noise expectation is zero),
- the mean ν and covariance matrix Σ of vectors $\mathbf{V}_i$, which are assumed i.i.d and Gaussian.

The model is entirely based on the finite-element representation defining both the template and deformations. The structure of this representation is characterized by a set of vertices and finite-elements. The number of vertices and the definition of finite elements can also be viewed as model parameters. In this paper, these structure parameters will however be estimated and set once for all before computations. The neighborhood structure on the vertex set defined in Section 2.4 could also be viewed as a model parameter and be set differently. In this paper, this structure is however fixed by a definition.

The observation model we formulated above can be viewed as a non-linear mixed effect model []: unobserved displacements $\mathbf{V}_i$ associated to each image can be interpreted as a random effect, and positions ν as a fixed effect. Due to the definition of the template, the regression of variables Y_{ij} on descriptive variables x_n is non-linear.

For the ease of presentation, we now write the model with matrix notations. Let us denote by $Y = (Y_{in})_{in}$ the vector of gray-level values of the n_t images, by $B = (B_{in})_{in}$ the residual errors, and by $V = (V_{ij})_{ij}$ the vector of vertex positions. Let us define for all $n = 1, \cdots, n_g$ the random matrix $W_n(V)$ of size $n_v \times n_e$ which has the random variable $\mathbb{1}_{x_n \in \mathcal{P}(V_{ij}, j \in \mathcal{N}_e)}$ on the i^{th} row and the e^{th} column. Let us further introduced the random matrix defined $W(V) = (W_1(V)|\cdots|W_{n_g}(V))$ defined by concatenation of matrices $W_n(V)$. Using these notations, the model formulation becomes

$$\begin{aligned} Y &= W(V)\rho + B, \\ B &\sim \mathcal{N}(0, \sigma^2 I), \\ V_i &\sim \mathcal{N}(\nu, \Sigma), \quad i = 1, \ldots, n_t. \end{aligned} \tag{10}$$

2.6. **Markovian properties.** In Section 2.4, we gave a graph structure to the vertex set $\mathcal{N}$ by defining neighborhood relationships between vertices. In this section, we focus on properties of the random field $\mathbf{V}_i|\mathbf{Y}_i$ which at time t_i describes vertex positions given observations and is defined on the graph $\mathcal{N}$. We show that this random field is Markovian for the graph structure. As we will see in Section 3.4, this Markovian property is of particular interest for the simulation of the random vector $\mathbf{V}_i|\mathbf{Y}_i$ in the SAEM algorithm.

By the Hammersley-Clifford theorem [**?**], a random field is Markovian on a given graph if and only if its distribution has a Gibbsian form on this graph. But, using a Bayes formula, we can express the distribution of $\mathbf{V}_i|\mathbf{Y}_i$ in terms of the conditional distribution of observations and the prior distribution of vertex displacements:

$$p(\mathbf{v}_i|\mathbf{y}_i;\theta) = \frac{p(\mathbf{y}_i|\mathbf{v}_i;\theta)p(\mathbf{v}_i;\theta)}{\int p(\mathbf{y}_i|\mathbf{v}_i';\theta)p(\mathbf{v}_i';\theta)d\mathbf{v}_i'},\ \forall\, i = 1,\cdots,n_t.$$

Hence, so as to prove that the random field $\mathbf{V}_i|\mathbf{Y}_i$ is Markovian, it suffices to show that the conditional distribution and the prior distribution are both Gibbsian for the graph $\mathcal{N}$.

Because the random vector $\mathbf{v}_i$ was assumed Gaussian, the prior distribution is of the form $p(\mathbf{v}_i;\theta) \propto \exp(-\frac{1}{2}(\mathbf{v}_i - \nu)^t\Gamma(\mathbf{v}_i - \nu))$, where Γ denotes the inverse of the covariance matrix Σ. Due to assumptions on local dependancies of vertex positions, this distribution can further be written as

$$p(\mathbf{v}_i;\theta) \propto \exp\left(-\frac{1}{2}\sum_{j=1}^{n_v}\sum_{k\in\mathcal{N}_j}(v_{ij}-\nu_j)^t\Gamma_{j,k}(v_{ik}-\nu_k) + (v_{ij}-\nu_j)^t\Gamma_{j,j}(v_{ij}-\nu_j)\right),$$

where $\mathcal{N}_j$ denotes the neighbors of the vertex j and $\Gamma_{j,k}$ the sub-matrix associated to vertices j and k. Thus, the prior distribution is of the form

$$p(\mathbf{v}_i;\theta) \propto \exp\left(\sum_{j=1}^{n_v}\sum_{k\in\mathcal{N}_j} q_{jk}(v_j,v_k) + \sum_{j=1}^{n_v} q_j(v_j)\right),$$

where the definition of potentials q_{jk} only involves pairs of neighbor vertices (v_j, v_k) (cliques of order 2) and the one of potentials q_j only requires singletons v_j (cliques of order 1). Consequently, the form of the prior distribution is Gibbsian. Note that this implies that the random vector $\mathbf{V}_i$ is Markovian.

Using the definition of the observation model in Equation (10), we have

$$p(\mathbf{y}_i|\mathbf{v}_i;\theta) \propto \exp\left(-\frac{1}{2\sigma^2}\sum_{n=1}^{n_g}(y_{in} - \sum_{e=1}^{n_e} W_n(v)_{i,e}\rho_e)^2\right).$$

This equation can also be written as

$$p(\mathbf{y}_i|\mathbf{v}_i;\theta) \propto \exp\left(-\frac{1}{2\sigma^2}\sum_{e=1}^{n_e}\sum_{n=1}^{n_g} W_n(v)_{i,e}(y_{in}-\rho_e)^2\right).$$

But, by definition, $W_n(v)_{i,e}$ only depends on v_{ij} for $j \in \mathcal{N}_e$. Hence, the conditional distribution is of the form

$$p(\mathbf{y}_i|\mathbf{v}_i;\theta) \propto \exp\left(-\frac{1}{2\sigma^2}\sum_{e=1}^{n_e}\tilde{q}(v_{ij}, j\in\mathcal{N}_e)\right).$$

But, the three vertices of a finite element p_e (indexed in the set $\mathcal{N}_e$) are pairwise-neighbors by definition of the graph structure. Hence, potentials $\tilde{q}_j$ are defined on cliques of order 3 and the conditional distribution $p(\mathbf{y}_i|\mathbf{v}_i;\theta)$ has a Gibbsian form.

We have shown that the random field $\mathbf{V}_i|\mathbf{Y}_i$ is Markovian on the graph $\mathcal{N}$. The Markovian property implies that the conditional distribution of any vertex position v_{ij} given the other positions only depends on positions in the neighborhood of j:

$$p\left(v_{ij}|v_{ik}, k\neq j, y; \theta^{(k-1)}\right) = p\left(v_{ij}|v_{ik}, k\in\mathcal{N}_j, y; \theta^{(k-1)}\right).$$

Using Hammersley-Clifford theorem, we can also give the form of this conditional probability:

$$\begin{aligned}(11)\ p\left(v_{ij}|v_{ik}, k\neq j, y; \theta^{(k-1)}\right) \propto \\ \exp\left(-\sum_{k\in\mathcal{N}_j}(v_{ij}-\nu_j)^t\Gamma_{j,k}(v_{ik}-\nu_k) - \frac{1}{2}(v_{ij}-\nu_j)^t\Gamma_{j,j}(v_{ij}-\nu_j)\right) \\ \times \exp\left(-\frac{1}{2\sigma^2}\sum_{e\in\mathcal{P}_j}\sum_{n=1}^{n_g} W_n(v)_{i,e}(y_{in}-\rho_e)^2\right),\end{aligned}$$

where $\mathcal{P}_j$ denotes the set of finite elements which the vertex j belongs to.

3. Estimation

In the next section, we will present the construction of a SAEM algorithm for the estimation of parameters in the model defined by Equation (9). We keep the same notation as before. By convention, we will use capital letters for random variables or vectors, lower case letters for either a realization of a random variable or a deterministic variable, and greek letters for parameters. In the context of [3], Y is the observed data, V is the missing data (or unobserved data), (Y, V) is the complete data and $\theta = (\rho, \sigma^2, \nu, \Sigma)$ are the set of all model parameters.

3.1. **The statistical problem.** In the statistical framework defined by Equation (10), we focus on the problem of the estimation of model parameters. Due to the non-linearity of the regression function, the posterior distribution $p(v|y;\theta)$ is only known up to a constant.

unobserved variables. Estimation problems with unobserved data has been early investigated by to the studied framework is the presence of unobserved variables. In the literature, the estimation problem with

premiere difficulte, variables non observee. EM. principe. deuxieme difficulte. pas d'expression explicite de la log-likelihood. conditional modes, sensible au bruit. SAEM, stochastic approximation. troisieme difficulte, MCMC pour echantilloner

In [2], Delyon and colleagues introduced another version of the EM algorithm, in which

the $Q^{(k)}(\theta)$ integral is evaluated by a stochastic approximation procedure. **********A REVOIR**********

3.2. **SAEM.** The classical EM algorithm introduced in [3] is based on an iterative procedure which aims at the maximization of the likelihood $p(y, V;\theta)$ of the complete data with respect to parameters θ. At each iteration k, the EM is decomposed into two successive steps called the expectation step (M-step) and the maximization step (M-step):

E-step: The log-likelihood of the complete data is evaluated by computing explicitely its expectation $Q^{(k)}$ conditional to observations y and a current approximation $\theta^{(k-1)}$ of parameters:

$$\forall\, \theta,\ Q^{(k)}(\theta) = \mathbb{E}(\log p(y, V;\theta)|y;\theta^{(k-1)}).$$

M-step: Parameter estimates are updated by

$$\theta^{(k)} = \arg\max_{\theta} Q^{(k)}(\theta),$$

which are the most likely parameter values for the current approximation of the log-likelihood.

The EM algorithm cannot be applied whenever the conditional expectation $Q^{(k)}$ cannot be explicitely computed in the E-step. For dealing with such a situation, Delyon, Lavielle and Moulines proposed to use a stochastic approximation of $Q^{(k)}$ based on a sampling of the posterior distribution [2]. In their algorithm (called SAEM), the traditional E-step of the EM algorithm is replaced by two new steps, a simulation step (S-step) and a stochastic approximation step (SA-step):

S-step: A non-observed data $v^{(k)}$ is sampled from the posterior distribution $p(v|y;\theta^{(k-1)})$.

SA-step: The conditional log-likelihood is approximated by

$$\hat{Q}^{(k)}(\theta) = (1-\gamma_k)\hat{Q}^{(k-1)}(\theta) + \gamma_k \log(p(y, v^{(k)};\theta^{(k-1)})),$$

using a weighted sum of both the previous approximation $\hat{Q}^{(k-1)}$ and another approximation obtained from the new sample $v^{(k)}$. In this approximation scheme, the weight γ_k is a scalar value between 0 and 1 which decreases to 0.

In [2], the convergence of the SAEM algorithm was proved under general conditions when the likelihood $p(y, V; \theta)$ belongs to a regular curved exponential family

$$\log p(y, V; \theta) = -\Psi(\theta) + \langle S(y, V), \Phi(\theta) \rangle, \tag{12}$$

where $\langle ., . \rangle$ is the scalar product and $S(y, V)$ is the minimal sufficient statistic of the model. For such a model, we have

$$Q^{(k)}(\theta) = -\Psi(\theta) + \langle s^{(k)}, \Phi(\theta) \rangle,$$

where $s^{(k)}$ denotes the conditional expectation $\mathbb{E}(S(y, V)|y; \theta^{(k-1)})$ at the k^{th} iteration. Hence, the SA-step of SAEM reduces to the approximation of $s^{(k)}$ and can be simplified as follows.

SA-step (curved exponential model): $s^{(k)}$ is updated by the stochastic approximation scheme

$$s^{(k)} = (1 - \gamma_k) s^{(k-1)} + \gamma_k S(y, v^{(k)}). \tag{13}$$

For the same kind of models, the M-step, which follows the S and SA steps in SAEM, is

M-step (curved exponential model): $\theta^{(k)}$ is updated by

$$\theta^{(k)} = \arg\max \left(-\Psi(\theta) + \langle s^{(k)}, \Phi(\theta) \rangle \right). \tag{14}$$

The application of the S-step of SAEM requires a sampling from the posterior distribution. In many situations, this distribution cannot however be sampled directly. For dealing with such a situation, Kuhn and Lavielle proposed to couple SAEM with a Monte Carlo Markov Chain scheme (MCMC) [4]. In their algorithm, the S-step definition at the k^{th} iteration is based on the construction of a non-homogeneous Markov chain with $p(V|y; \theta^{(k-1)})$ as unique stationary distribution. The convergence of the combined SAEM/MCMC algorithm was proved under the following general conditions [**?**].

C1: For any $\theta \in \Theta$, the MCMC algorithm generates a uniformly ergodic chain for which the invariant probability is $p(V|y; \theta)$.

C2: For all k in $\mathbb{N}$, $\gamma_k \in [0, 1]$, $\sum_{k=1}^{\infty} \gamma_k = \infty$ and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$.

More recently, the convergence was also proved by Allassonniere under more general conditions [].*****ADELINE***** petite biblio ici.

3.3. **Application of SAEM.**

3.3.1. *Model Family.* The log-likelihood of the complete data (y, v) in the model described by Equation (10) can be expressed as

$$\begin{aligned} \log p(y, v; \theta) &= \log p(y|v; \theta) + \log p(v; \theta), \\ &= -\frac{1}{2\sigma^2}(y - \mathcal{W}(v))^t(y - \mathcal{W}(v)) - \frac{1}{2}\sum_{i=1}^{n_t} \mathbf{v}_i^t \Sigma^{-1} \mathbf{v}_i - K(\theta), \end{aligned} \tag{15}$$

where $K(\theta)$ is the logarithm of the normalization constant of the probability distribution $p(y, v; \theta)$. This equation can further be written as

$$\log p(y, v; \theta) = \sum_{i=1}^{3} S_i(v)\Phi_i(y, v) - \Psi(\theta),$$

with

$$\begin{array}{lll} S_1(v) = \frac{1}{n_t}\sum_{i=1}^{n_t} \mathbf{v}_i, & S_2(v) = \frac{1}{n_t}\sum_{i=1}^{n_t} \mathbf{v}_i\mathbf{v}_i^t, & S_3(v) = \mathcal{W}(v). \\ \Phi_1(\theta) = -\frac{n_t}{2}\Sigma^{-1}\nu, & \Phi_2(\theta) = & \Phi_3(\theta) = . \end{array}$$

Therefore, the model we study belongs to the curved exponential family defined by Equation (12). Moreover, the sufficient statistics associated to the model are $S_1(V)$, $S_2(V)$ and $S_3(V)$. Adeline********** Peut tu montrer precisement comment on se ramene a l'Equation + verifier les equations qui suivent car j'ai fait bcp de modif.

3.3.2. *SA-step.* Consequently, when applying SAEM to our model, the SA-step at the k^{th} iteration simply consists of computing

$$s_1^{(k)} = (1 - \gamma_k)s_1^{(k-1)} + \frac{\gamma_k}{n_t}\sum_{i=1}^{n_t} \mathbf{v}_i^{(k)}, \tag{16}$$

$$s_2^{(k)} = (1 - \gamma_k)s_2^{(k-1)} + \frac{\gamma_k}{n_t}\sum_{i=1}^{n_t} \mathbf{v}_i^{(k)}\mathbf{v}_i^{(k)t}, \tag{17}$$

$$s_3^{(k)} = (1 - \gamma_k)s_3^{(k-1)} + \gamma_k\mathcal{W}(v^{(k)}). \tag{18}$$

3.3.3. *M-step.* In the M-step, parameters ρ, ν, σ^2 and Σ are to be estimated in the M-step as follows.

$$\rho^{(k)} = (s_3^{(k)t}s_3^{(k)})^{-1}s_3^{(k)t}y, \tag{19}$$

$$\sigma^{2(k)} = \frac{1}{n_t n_g}(y - s_3^{(k)}\rho^{(k)})^t(y - s_3^{(k)}\rho^{(k)}), \tag{20}$$

$$\nu^{(k)} = s_1^{(k)}, \tag{21}$$

$$\Sigma^{(k)} = s_2^{(k)} - \nu^{(k)}\nu^{(k)t}. \tag{22}$$

**********Adeline, peut-on le montrer? ou tout au moins en donner une explication. car cela ne me semble pas evident.

Let us mention that Equations (19) and (20) are nothing else than usual equations for the estimation of the regression vector and noise variance in the linear model $y = s_3^{(k)}\rho + b$. This regression model corresponds exactely to model (10) when V is fixed and equal to $v^{(k)}$. Besides, Equations (21) and (22) are usual equations for the estimation of the mean and covariance of a Gaussian random vector from a set of independant realizations $v_i^{(k)}$, for $i = 1, \cdots, n_t$.

The S-step of SAEM also requires the inverse matrix Γ of the matrix Σ to be estimated. But, the matrix $\Sigma^{(k)}$ is ill-conditioned and the computation of its inverse is numerically unstable. **********donner une explication ici, on met un a priori la ou il n'y a pas de mvt apparent.********** Therefore, we decide to sidestep this conditioning problem by substituting $\Sigma^{(k)}$ with $\Sigma^{(k)} + \epsilon I_{2n_g}$, where I_{2n_g} is the identity matrix of size $2n_g \times 2n_g$. **********il faudrait parler de la projection. Moreover...

Finally, we obtain an estimator of Γ at iteration k of SAEM by computing

$$\Gamma^{(k)} = \text{Proj}\left((\Sigma^{(k)} + \epsilon I_{2n_g})^{-1}\right). \tag{23}$$

3.4. **S-step.** In the S-step of the k^{th} iteration, we deal separately with each time t_i, since vertex positions are assumed to be time-independant in the model. For each time t_i, we scan successively all vertices and for each vertex j, we update the vertex position V_{ij} by simulating in the target distribution

$$\pi_{ij}^{(k)}(v_{ij}) = p\left(v_{ij} | \{v_{il}^{(k-1)}, l < j\}, \{v_{il}^{(k)}, j < l\}, y; \theta^{(k-1)}\right). \tag{24}$$

This distribution is the local conditional distribution at vertex j and time t_i of the random vector $V|y$. Its expression is given in Equation (11). In Section 2.6, we have shown that the random field $V|y$ is Markovian. Hence, the target distribution depends only on positions of vertices in the neighborhood of j. This means that the simulation of each V_{ij} involves only a few vertices located in the neighborhood of the vertex j.

The simulation of V_{ij} is done using a Metropolis-Hastings scheme (MH). This MH consists of three steps:

MH1: A candidate position v'_{ij} is sampled from a proposal distribution $q_{ij}^{(k)}$, as described next.

MH2: We compute the so-called acceptance probability

$$\alpha = \min\left(\frac{\pi_{ij}^{(k)}(v'_{ij})}{\pi_{ij}^{(k)}(v_{ij}^{(k-1)})}\frac{q_{ij}^{(k)}(v_{ij}^{(k-1)})}{q_{ij}^{(k)}(v'_{ij})}, 1\right)$$

using Equations (11) and (24).

MH3: We simulate u with an uniform distribution $\mathcal{U}[0,1]$. If $u < \alpha$ then $v_{ij}^{(k)} = v'_{ij}$ else $v_{ij}^{(k)} = v_{ij}^{(k-1)}$.

The definition of the proposal distribution q_{ij} in the step **MH1** is crucial for the speed of convergence of the whole SAEM. We define this distribution carefully

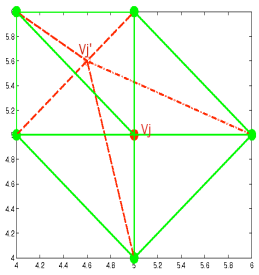

FIGURE 5. A refaire

so as to ensure that candidate positions are reasonable. As illustrated on Figure 5, we define the set $c_{ij}^{(k)}$ of points in the interior of the polygon formed by positions v_{il} of neighbor vertices. The new candidate position v'_{ij} is chosen in the set $c_{ij}^{(k)}$, excluding points which would generate mesh singularites (crossings of triangle edges). The selection of the position v'_{ij} is done randomly using the uniform distribution $q_{ij}^{(k)}$ on non-singular points of the set $c_{ij}^{(k)}$. Notice that this distribution does not depend on the current vertex position $v_{ij}^{(k-1)}$ and that $q_{ij}^{(k)}(v'_{ij}) = q_{ij}^{(k)}(v_{ij}^{(k-1)})$. Hence the acceptance probability reduces to

$$\alpha = \min\left(\frac{\pi_{ij}^{(k)}(v'_{ij})}{\pi_{ij}^{(k)}(v_{ij}^{(k-1)})}, 1\right)$$

The MH algorithm we constructed can be viewed as a random walk scheme.

4. RESULTS

4.1. **Description of the data.**

4.2. **Initialization.** initialisation du template. description de la methode. details d'implementation: combien de parametres, dimension des images.

The sequence $(\gamma_k)_{k\geq 0}$ has to be chosen to fulfill the condition (C2) (see Section 1). We recommend to use $\gamma_k = 1$ during the first iterations $1 \leq k \leq K$, and $\gamma_k = (k-K)^{-1}$ during the last iterations. Indeed, the initial guess $\theta^{(0)}$ may be far from the maximum likelihood value and the first iterations with $\gamma_k = 1$ allow to converge to a neighborhood of the maximum likelihood estimate. ********BOF, on se contredit un peu la.***** Furthermore, the inclusion of a hybrid Gibbs procedure (instead of a Metropolis-Hastings procedure in the classic SAEM algorithm) slows up the convergence of the extended SAEM algorithm. Therefore, to ensure the convergence of the Markov Chain, we implement the extended SAEM with $K = 100$ iterations. Then smaller step sizes during 400 additional iterations ensure the almost sure convergence of the algorithm to the maximum likelihood estimate. We implement the SAEM

algorithm in a Matlab function. code available for a public use. It takes about XXX minutes for the SAEM algorithm to converge with 500 iterations for a registration problem of 10 images, on a conventional Intel Pentium IV 2.8 GHz workstation.

4.3. **Application.** template initial/final. (image+maillage)
difference pre/post recalage entre observations et template
convergence des estimateurs.

interet de l'estimation: debruitage, on obtient un template, on peut a partir d'une inversion de l'equation (1) retrouver des proprietes physique de l'objet. propager cette information a toutes les observations par le biais des deformations. on peut evaluer les proprietes des deformations apparentes dans l'image, different des deformations reelles mais proche.

In images such as Xray scans, gray-level values essentially depend on the chemical composition of the object. Each chemical component (fat, air, bone, water,...) is observed within a known range of possible gray-level values, called the Hounsfield unit.

4.4. **A numerical study.** montrer des exemples de simulation.

cause des erreurs d'estimation des deplacements et des parametres = depl estime ne sont pas deplacements reels, aperture problem.

TABLE 1. Estimation of parameters on synthetic data.

				bias	variance
param.	unit	size	n_t	Av.±Std.(Max.)	Av.±Std.(Max.)
ρ	gray-level	1069	5	-1. ± 81 (456)	6 ± 3 (30)
			10	-2 ± 82 (459)	5 ± 4 (78)
σ	gray-level	1	5	3.5 ± 0.3 (3.9)	- ± - (-)
			10	3.8 ± 0.3 (4)	- ± - (-)
ν	pixels	1106	5	0.2 ± 1.4 (15.1)	0.2 ± 0.1 (1)
			10	0.2 ± 1.4 (15.4)	0.19± 0.1 (1)
Σ	pixels	15180	5	0. ± 0. (0.35)	0. ± 0. (0.01)
			10	0. ± 0. (0.34)	0. ± 0. (0.02)

5. DISCUSSION

The representation of the template by a piecewise-constant function is a way to simplify the image. depend on the complexity of images. but in our application it can

Let us point out here that the affine representation of deformations is defined with the same finite elements used for describing the template gray-level values. Such an approach differs significantly from the one of Allassonniere et al. who have two distinct setups for template and deformations. We believe that having

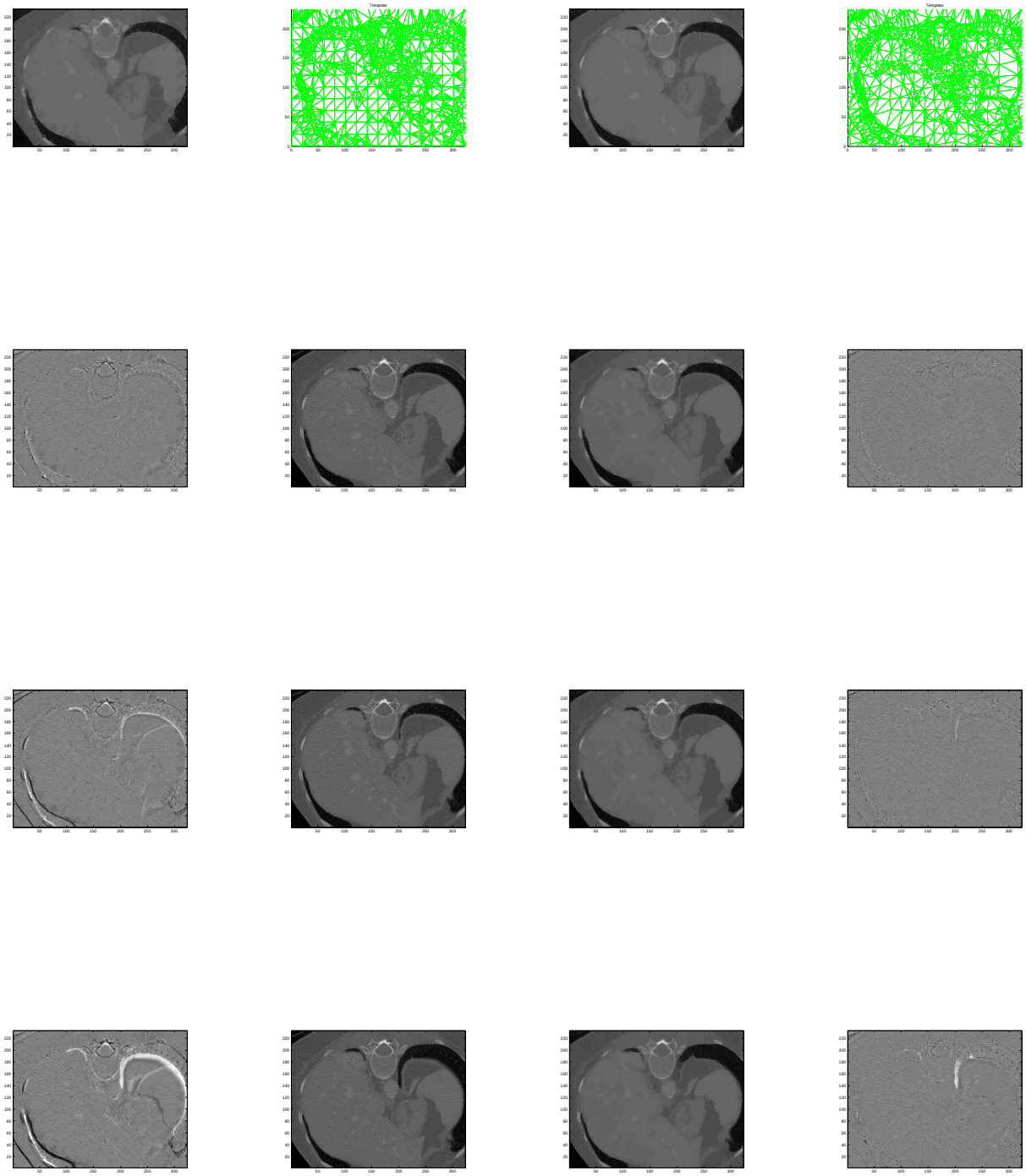

FIGURE 6. example caption

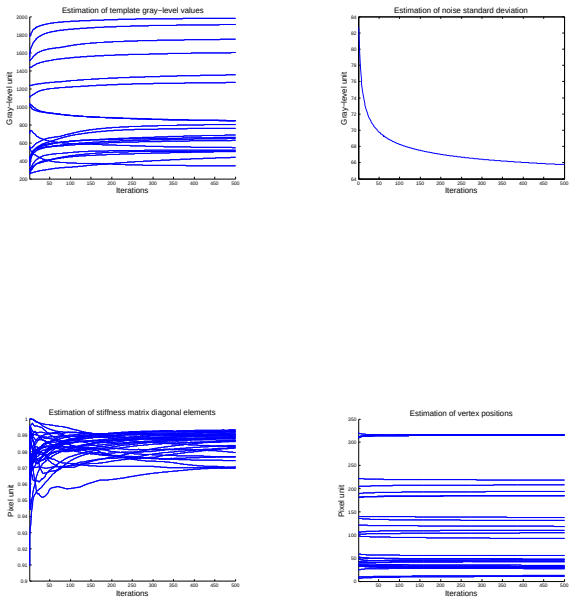

FIGURE 7. example caption

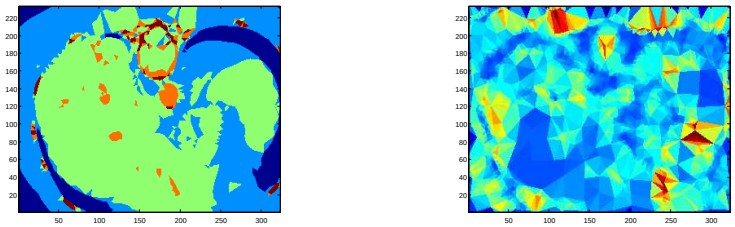

FIGURE 8. segmentation

a joint finite-element setup for both the template and deformations is relevant for applications which involve high-dimensional images. It is more compact and parcimonious than a separate setup.

points forts de la modelisation: representation du template par fonction constante par morceaux. permet d'adapter le modele a la geometrie de le l'objet dans l'image. reduit la complexite. par ailleurs, la representation affine des deformation est elle meme definie a partir de la decomposition en element finis utilisee pour template. representation plus compacte et parcimonieuse, essentielle pour l'application a de grandes images. crucial pour l'extension en 3D.

points forts: modelisation statistique du recalage de sequence d'images, adaptation des methodes SAEM avec methode hybride Gibbs/MH pour simulation.

limites: estimation du mvt apparent. modelisation physique de la deformation : passer d'une estimation du mouvement apparent a estimation du mouvement reel. overfitting. identifiabilite des parametres

ameliorations: estimation du graphe de voisinage. raffinement de maillage. MCMC adaptative metropolis. autres modele de bruit possible. pour d'autres applications, correlation au cours du temps des deplacements des noeuds

When combining a MCMC to SAEM in the S-step, the choice of the MCMC is a critical issue: although the convergence of the algorithm is generally ensured, the speed of convergence strongly depends on the MCMC.

discuter davantage du global vs local. plus du fait que d'un point de vue local nous avons des des proprietes markovienne The use of this Markovian structure for the simulation under the target distribution obviously improves the efficacy of the MCMC procedure. However, the direct simulation of V in this multi-dimensional distribution is difficult, even with a Metropolis-Hastings (MH) procedure. Hence, we use a Metropolis-Hastings-Within-Gibbs (MHWG) approach, in order to simulate each component V_{ij} of V successively, the simulation in a one-dimensional distribution being more manageable. Let us now give the details of iterations of the MHWG.

****

References

[1] S. Allassonniere, Y. Amit, and A. Trouve. Towards a coherent statistical framework for dense deformable template estimation. *J. R. Stat. Soc.: Series B (Statistical Methodology)*, 69(1):3–29, Feb 2007.

[2] B. Delyon, M. Lavielle, and E. Moulines. Convergence of a stochastic approximation version of the EM algorithm. *Ann. Statist.*, 27:94–128, 1999.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B*, 39:1–38, 1977.

[4] E. Kuhn and M. Lavielle. Coupling a stochastic approximation version of EM with a MCMC procedure. *ESAIM Probab. Stat.*, 8:115–131, 2004.

Laboratory MAP5, University Paris Descartes, CNRS UMR 8145, 45, rue des Saints-Pères, 75270 PARIS cedex 06 FRANCE.

E-mail address: [richard,samson]@math-info.univ-paris5.fr