Geodesic Shooting and Diffeomorphic Matching Via Textured Meshes

Stéphanie Allassonnière[1], Alain Trouvé[2], and Laurent Younes[3]

[1] LAGA, Institut Galilée, University Paris 13, France
Stephanie.Allassonniere@cmla.ens-cachan.fr
[2] CMLA, Ecole Normale Supérieure, Cachan, France
Alain.Trouve@cmla.ens-cachan.fr
[3] CIS, Johns Hopkins University, Baltimore MD
laurent.younes@jhu.edu

Abstract. We propose a new approach in the context of diffeomorphic image matching with free boundaries. A region of interest is triangulated over a template, which is considered as a grey level textured mesh. A diffeomorphic transformation is then approximated by the piecewise affine deformation driven by the displacements of the vertices of the triangles. This provides a finite dimensional, landmark-type, reduction for this dense image comparison problem. Based on an optimal control model, we analyze and compare two optimization methods formulated in terms of the initial momentum: direct optimization by gradient descent, or root-finding for the transversality equation, enhanced by a preconditioning of the Jacobian. We finally provide a series of numerical experiments on digit and face matching.

1 Introduction

The theory of deformable templates [10, 4, 3] provides a large range of applications to pattern and shape analysis and matching, with specific important achievements in object recognition and medical imaging. The large deformation diffeomorphic approach, initiated in [18, 6], has proved particularly accurate and robust in this framework. Several algorithms have been developped, ranging from landmark matching [13, 5, 1, 9, 7, 14] to images [16, 2], shape matching via measures [8] or currents [20]. These algorithms come with a strong theoretical support, regarding their well-posedness [6, 18, 19], and their properties, in terms of metric distances [23, 16], and in relation to infinite dimensional mechanics, yielding the notion of conservation of momentum and its normality [15, 21, 11]. As noticed in [21], this can also be embedded in a Hamiltonian, or optimal control, framework. We shall adopt this last point of view in the present paper.

Assume that a template and a target images are given. Assume also that a region of interest is extracted from the template, on which a triangulation is overlayed, resulting in a textured mesh. We shall develop a dense matching algorithm which computes a piecewise affine deformation between the images. This deformation is controlled by a dynamical evolution of the vertices of the

A. Rangarajan et al. (Eds.): EMMCVPR 2005, LNCS 3757, pp. 365–381, 2005.

triangulation (through an ordinary differential equation), which will end-up in a formulation closely related to diffeomorphic landmark matching [13, 5, 21]. Because of this, we will henceforth refer to the vertices of the triangulation as landmarks, although they do not need to correspond to any point of interest within the images.

From the evolution of the landmark, we will deduce an evolution of the triangulation, and build from it a piecewise affine deformation. The quality of the matching is measured by a data term based on the mean squared error between the deformed template and the target within the region of interest covered by the triangulation. The whole procedure is therefore governed by the ordinary differential equation (ODE) satisfied by the landmarks, which will be specified in term of a non-autonomous (time-dependent) vector field on the image plane. This vector field can be seen as a control for the final matching, and its cost will be defined as an integrated measure of smoothness of the vector field along time.

The problem can be handled by an optimal control (or Hamiltonian) approach, which, thanks to the maximum principle, can be parametrized by what is called the initial momentum, which evolves through a conservation equation and allows to recover the ODE and the deformation. In our context, this point of view has been introduced in [15] and used in [21] for landmark matching, using gradient descent algorithms. We will here adapt the gradient descent algorithms to our image matching framework, and analyze an alternative optimization method, also applicable to standard landmark matching, called shooting in the optimal control literature. This is a root-finding method (using Newton's algorithm), designed to solve the transversality equation associated to the problem.

The paper is organized as follows. We start with describing a generic landmark based matching problem in terms of optimal control, first as an infinite dimensional problem, and then reduce it to finite dimensions, using usual arguments of the theory of smoothing splines. We then describe our approaches for solving this problem: direct minimization by gradient descent and root-finding by Newton's method. This last method will be briefly illustrated by landmark matching examples. We will then focus on our image matching problem, introducing notation and computing the elements needed for the two algorithms. The paper will end with a presentation of some experiments with 2D images.

We first fix notation. Images are assumed to be defined on Ω, an open bounded set of $\mathbb{R}^n$ with regular boundary (piecewise C^1). We assume that a template image (denoted I_0) has been selected, and that a triangulation has been overlayed on the template, and denote $(x_1, ..., x_N)$ the vertices of the triangulation. Typically, $(x_1, ..., x_N)$ are chosen first, as landmarks, and the triangulation is deduced, using in our case Delaunay's triangulation. We denote by x_i^d the d^{th} coordinate of the vector x_i. The landmarks will serve as control points to estimate a diffeomorphism ϕ which will provide a dense matching between I_0 and a target image I_1.

For vectors x, y, the notation $\langle x, y\rangle$ will be used for the standard dot product $x^T y$. For dot products on a Hilbert space V, the notation $\langle x, y\rangle_V$ will be used.

2 Optimal Control Problem

2.1 Context

We provide a Hamiltonian formulation of the landmark matching large deformation setting, originally introduced in [13]. The interpretation already appeared in [15], [21], and can be summarized as follows. The evolution of the landmarks is driven by a single non-autonomous ODE $dy/dt = v_t(y)$. This defines N landmark trajectories, denoted $t \mapsto q_i(t), i = 1, \ldots, N$, each satisfying the system

$$\begin{cases} \dfrac{dq_i(t)}{dt} & = v_t(q_i(t)) \\ \\ q_i(0) & = x_i \ . \end{cases} \tag{1}$$

Here, $(t, y) \mapsto v_t(y)$ is a time dependent velocity vector field, which serves as a control variable for our system of N landmarks.

As done in the optimal control theory for image matching, developped among others by Dupuis et al. ([6]), we introduce an energy which has to be minimized under constraints. This energy stems from a tradeoff between a deformation constraint and a data attachment term. The deformation term is equal to the integration over time (between 0 and 1) of the kinetic energy of the transformation.

The instantaneous kinetic energy is defined as the norm $\|v_t\|_V^2/2$ of the velocity field introduced in (1). The total energy is $E_k(v) = \frac{1}{2}\int_0^1 ||v_t||_V^2 dt$. This norm is a Hilbert norm (defined on a Hilbert space V); it is designed to ensure that v_t is sufficiently smooth. For this purpose, V is assumed to be continuously embedded in $C_0^1(\Omega)$, the set of continuously differentiable functions which vanish on the boundary of Ω. Because of this, V is a so-called self-reproducing kernel Hilbert space, which implies that there exists a kernel k_V, defined on $\Omega \times \Omega$, taking values in the set of symmetric (n, n) matrices, such that: (i) for all $x \in \Omega$, and for all $\alpha \in \mathbb{R}^n$, the vector field $k_V(x)\alpha : y \mapsto k_V(x, y)\alpha$ belongs to V and (ii) $\langle k_V(x)\alpha, w\rangle_V = \langle w(x), \alpha\rangle_{\mathbb{R}^n}$, for all $w \in V$.

If a set of landmarks: $q = (q_1, \ldots, q_N)$ is given, we denote by $K(q)$ the $nN \times nN$ matrix consisting on the $n \times n$ blocks $k_V(q_i, q_j)$: $K(q) = (k_V(q_i, q_j)_{1\leq i,j\leq N})$.

We assume that the data attachment term only depends on the final configuration of the landmarks: $q(1)$, and of other constants of the problem (in our case: the template and target images I_0 and I_1). We will denote it by $g_{I_0,I_1}(q(1))$, or simply $g(q(1))$ if there is no ambiguity on the compared images. This will be detailed in section 3 for our image comparison algorithm. However, since most of the developments can be done by only assuming that $q \mapsto g(q)$ is twice differentiable, we carry on this discussion assuming a generic data attachment term satisfying this property.

With this notation, introducing a positive weigth λ, the complete energy is

$$E(v, q(1)) = \frac{1}{2}\int_0^1 ||v_t||_V^2 dt + \lambda g(q(1))\,. \tag{2}$$

Remark 1. *The dynamical aspect of the formulation can be compared to linear smoothing spline approaches, which will essentially remove the time variable, using a single $v \in V$, and replace* (1) *by $q_i(1) = q_i(0) + v(q_i(0))$, with the integral in the energy term replaced by $\|v\|_V^2$. As already demonstrated in [13, 5], our formulation ensures non-ambiguous and smoother deformation when interpolated to Ω, and is consistent with the constraint of building diffeomorphisms, which is not the case with linear splines.*

The smoothness assumptions on $(v_t, t \in [0,1])$ ensures existence and uniqueness of the solutions of the ODE, so that the landmarks $q(.)$ are defined at all times.

2.2 Reduction of Dimension

Standard arguments, similar to those used in the theory of smoothing splines, and relying on the kernel k_V of the Hilbert space V, allow to characterize the velocity field v_t by a finite dimensional time dependent system [22], [13]. In our case, this has an interesting Hamiltonian interpretation [21], which can also be derived from Pontryagin's maximum principle in optimal control [12]. The result is the existence at all times t of N vectors $p_i(t) \in \mathbb{R}^n$, such that:

$$v_t = \sum_{i=1}^{N} k_V(q_i(t)) p_i(t) \ . \qquad (\mathit{Interpolation\ Formula}) \tag{3}$$

The vector $p_i(t)$ is called the *momentum of the i^{th} landmark at time t*. The joint evolution of the landmarks and the momentum can be written in a standard Hamiltonian form for $H(q,p) = \frac{1}{2}\langle p, K(q)p\rangle$ (see Appendix)

$$\begin{cases} \dfrac{dq}{dt} = \dfrac{\partial H}{\partial p}(q,p) = K(q(t))p(t) \\[2ex] \dfrac{dp}{dt} = -\dfrac{\partial H}{\partial q}(q,p) = -\dfrac{1}{2}\nabla_{q(t)} K(p(t), p(t)) \end{cases} \tag{4}$$

where $\nabla_q K(p,p)$ is defined as follows. Let $d_q K$ be the differential of $q \mapsto K(q)$: since K is a matrix, the linear map $h \mapsto d_q K.h$ is matrix valued. We define $\nabla_q K(p,p)$ to be the vector w such that, for all $h \in \mathbb{R}^n$, $\langle (d_q K.h)p, p\rangle = \langle w, h\rangle$. From the definition of K, we have $H(q(t), p(t)) = ||v_t||_V^2/2$ and the Hamiltonian remains constant along the trajectories of (4), yielding

$$E_k(v) = \frac{1}{2}\int_0^1 ||v_t||_V^2 dt = \frac{1}{2}\langle K(q(0))p(0), p(0)\rangle \ . \tag{5}$$

Using system (4), the time evolution of the momentum and landmarks can be computed from the initial momentum and landmarks. In particular, since the initial position of the landmarks is fixed, their final position, $q(1)$, can be seen as a function of the initial momentum, $p(0)$, alone. According to this, our energy function can be seen as only depending on this initial momentum, a finite dimensional variable.

$$\mathcal{E}(p(0)) = \frac{1}{2}\langle K(q(0))p(0), p(0)\rangle + \lambda g(q(1)) \ . \tag{6}$$

Remark 2. *Because of the formula (3) we can reconstruct a global deformation by integrating the equation $dy/dt = v_t(y)$ with various initial conditions: this is the flow associated to the ODE, and provides a diffeomorphism on Ω which only depends on the initial momentum and initial landmarks, since this was the case for v_t. We will refer to it as the reconstructed diffeomorphism.*

Returning to our optimal control problem, the optimal trajectory must satisfy an **additional transversality condition** (see Appendix for a brief derivation and [8] for a more general case). This is given by

$$p(1) + \lambda \nabla_{q(1)} g = 0 \ . \tag{7}$$

Since $p(1)$ and $q(1)$ can be considered as functions of $p(0)$, this is a non-linear equation in the initial momentum.

We now analyze and describe two methods for the solution of our variational problem. The first one is to directly minimize the energy by gradient descent, with respect to the initial momentum $p(0)$. The second is to solve (7), again with respect to $p(0)$.

2.3 Algorithms

Gradient Descent. Several gradient descent algorithms which minimize the landmark-based energies with respect to the landmark trajectories have been developed in [13, 1, 5]. An algorithm working with the initial momentum has been proposed in [21], yielding the following gradient descent algorithm:

Algorithm 1. Gradient Descent on $p(0)$

```
Choose an initial $p(0)$, and $\delta \in \mathbb{R}_+^*$, then iterate until convergence:
    $p(0)^{new} = p(0)^{old} - \delta \nabla_{p(0)^{old}} \mathcal{E}$
where $\nabla_{p(0)}\mathcal{E} = K(q(0))p(0) + \lambda \left(\frac{\partial q(1)}{\partial p(0)}\right)^T \nabla_{q(1)} g$ .
```

Solving the transversality Equation. To solve (7), we use a variant of Newton's algorithm. The advantage of this algorithm is its convergence speed. Choosing an initial point in a neighborhood of the solution provides a quadratic convergence rate. This yields the following iterations : let $G(p(0)) = p(1) + \lambda \nabla_{q(1)} g$.

Here we have, denoting $d_q^2 g$ the Hessian matrix (second derivative) of g,

$$d_{p(0)}G = \frac{\partial p(1)}{\partial p(0)} + \lambda d^2_{q(1)} g \frac{\partial p(1)}{\partial p(0)} \, . \tag{8}$$

Algorithm 2. Newton's Algorithm on transversality Condition

```
Choose an initial $p(0)$, then iterate until convergence :
```

$$p(0)^{new} = p(0)^{old} - (d_{p(0)^{old}} G)^{-1} G(p(0)^{old})$$

However, Newton's method must be used with care, since its convergence is not guaranteed. It is sometimes a good idea to combine gradient descent and Newton's algorithm: use gradient descent as long as it is efficient (large variations of the energy), and switch to the second algorithm when it slows down (hopefully in a close neighborhood of a local minimum). Note however that such an approach was unnecessary in our handwritten digit and face experiments for which we could start directly with the root-finding algorithm and always achieve convergence.

There is an other issue in Newton's algorithm : to compute each iteration, we have to invert a matrix. Depending on its conditioning, the inversion could make the algorithm diverge. To avoid this issue, before the inversion, we pre-condition the matrix. The choice we made is to project the matrix on its main singular directions.The resulting vector p_r is an approximation of the real solution of (7) which converge when r increases. So that the resulting algorithm is :

Algorithm 3. Newton's Algorithm on Transversality Condition, Preconditioning

```
Choose an initial value of $p(0)$, then iterate until convergence
```

$$p_0^{k+1} = p_0^k - V^T D_r U^T G(p_0^k) \text{ where } [U\ S\ V] = svd\left(\frac{\partial p(1)}{\partial p(0)} + d^2_{q(1)} g \frac{\partial q(1)}{\partial p(0)}\right)$$

and $D_r = \mathrm{diag}(1/\lambda_1, \cdots, 1/\lambda_r, 0, \cdots, 0)$ where the λ_i's are the singular values of S sorted in decreasing order.

Variation of the Hamiltonian System. Both algorithms require the computation of the differential of the end-points of system (4) with respect to the initial momentum $p(0)$. This is obtained by differentiating the system, yielding a new evolution providing the required differentials.

$$\begin{cases} \frac{d}{dt}\left(\frac{\partial q(t)}{\partial p(0)}\right) = \frac{\partial K(q(t))}{\partial q(t)} \frac{\partial q(t)}{\partial p(0)} + K(q(t)) \frac{\partial p(t)}{\partial p(0)} \\ \frac{d}{dt}\left(\frac{\partial p(t)}{\partial p(0)}\right) = -\frac{\partial p(t)}{\partial p(0)} \frac{\partial K(q(t))}{\partial q(t)} p(t) - p(t) \frac{\partial}{\partial p(0)}\left(\frac{\partial K(q(t))}{\partial q(t)}\right) p(t) - p(t) \frac{\partial K(q(t))}{\partial q(t)} \frac{\partial p(t)}{\partial p(0)} . \end{cases} \tag{9}$$

Remark 3. *This additionnal transversality equation enables the use of Newton's algorithm which wouldn't have been so easy working only on the energy: running this algorithm to solve* $\nabla_{p(0)} E = 0$ *requires to compute* $\frac{d^2 q(1)}{dp(0)^2}$ *and so to differentiate twice then solve the Hamiltonian system (4).*

2.4 A First Application : Landmark Matching

As a first application of this framework, we discuss landmark matching: in this special case, the data attachment term is equal to the sum of squared distances between the final landmarks and the target landmarks $y = (y_i)_{1\leq i\leq N}$: $g(q(1)) = \sum_{i=1}^{N} ||q_i(1) - y_i||^2_{\mathbb{R}^n}$. In this case, the first and second derivatives of the data attachment term are easy to compute : $\nabla_{q(1)} g = 2\sum_{i=1}^{N} q_i(1) - y_i$ and $d^2_{q(1),q(1)} g = 2Id_{nN}$, Id_{nN} being the identity matrix in $\mathcal{M}_{nN}(\mathbb{R})$. This yields the two following algorithms :

Gradient descent: Choose an initial $p(0)$, and a constant δ, then iterate until convergence: $p(0)^{new} = p(0)^{old} - \delta(K(q(0))p(0) + 2\lambda \left(\frac{\partial q(1)}{\partial p(0)}\right)^T (q(1) - \mathbf{y}))$

Newton's method: Choose an initial value of $p(0)$, then iterate until convergence : $p(0)^{new} = p(0)^{old} - (\frac{dp(1)}{dp(0)} + 2\lambda Id_{nN})^{-1}(p(1) + \lambda(q(1) - \mathbf{y}))$

Figure 1 shows the results of Newton's Method for 2 sets of landmarks.

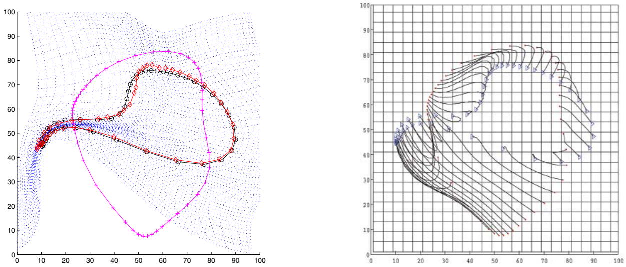

Fig. 1. Landmark matching : left : template (+), targets (∘), final landmarks (⋄) and deformation of the inherent space ; right : landmarks trajectories

3 Image Matching on Piecewise Affine Triangulations

We now focus on our primary application: image matching, which goes as follows. We start with a template image which has previously been annotated with landmarks. This will define a region of interest in the template which will then be warped to the target image so that it delimitates a region with similar content.

The region of interest is provided by a triangulation associated to the landmarks, for example, Delaunay's triangulation whose advantage is among others that no triangle is included in an other. For this particular case, this yields a convex region which is partitioned into triangles (or simplices in higher dimension), as illustrated in figures 2. We now define the data attachment term $g_{I_0,I_1}(q(1))$. Denote by $T_1, \ldots, T_r$ the family of triangles forming the partition of the region

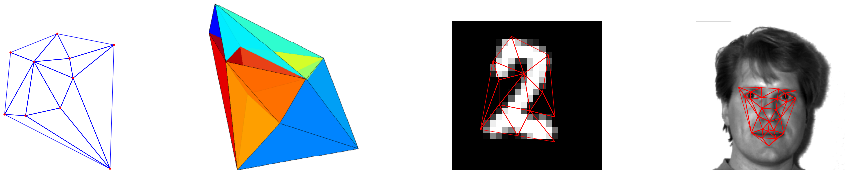

Fig. 2. Triangulation (2D), tessellation (3D), and examples of template triangulations

of interest in the template. Each triangle T_k have vertices from the initial landmarks, say $T_k = (x_{i_{k1}}, x_{i_{k2}}, x_{i_{k3}})$. The landmark evolution (4) displaces T_k into the triangle $T'_k = (q_{i_{k1}}(1), q_{i_{k2}}(1), q_{i_{k3}}(1))$ in the target. There exists a unique affine transformation ϕ_k which transforms T_k onto T'_k, and, assuming that the orientation of T'_k is consistent with the one of T_k, we define the piecewise affine homeomorphism

$$\phi : R := \bigcup_{k=1}^{r} T_k \mapsto R' := \bigcup_{k=1}^{r} T'_k \tag{10}$$

by $\phi_{|T_k} = \phi_k$. (Although this does not appear in the notation, ϕ depends on the landmark trajectories.) To keep the consistency of the triangle orientations, a sufficient condition is to choose the kernel variance according to the constant λ. (cf : Annexes) The data attachment term g is then defined by

$$g(q(1)) = \int_{R'} (I_0 \circ \phi^{-1} - I_1)^2 dy \ . \tag{11}$$

3.1 Reformulation of the Data Attachment Term

We now express g into a form which will simplify the computation of its derivatives (recall that we need the first derivative for gradient descent, and the second for Newton's method). First, introducing the triangulation, we have, with the notation above,

$$g(q(1)) = \sum_{k=1}^{r} \int_{T'_k} |I_1(y) - I_0 \circ \phi_k^{-1}(y)|^2 dy \ . \tag{12}$$

In order to lighten the notation, we only focus, from now, on the 2D case. Higher dimension is adressed with an identical argument (simply replacing triangles by simplices).

We can remove the dependence of the integration domain on ϕ by a change of variables yielding

$$g(q(1)) = \sum_{k=1}^{r} \int_{T_k} |I_1(\phi_k(x)) - I_0(x)|^2 |d_x\phi_k| dx \ . \tag{13}$$

Note that, because ϕ_k is affine, the jacobian is equal to the ratio between the surfaces of the target and template triangles, and will be easily handled in the computation of derivatives. We now make the computation explicit by introducing a local parametrization of the interior of each triangle.

Using our notation, each point in the interior of T_k is uniquely described by 2 coordinates (α, β), with $0 \le \alpha \le 1$, $0 \le \beta \le 1 - \alpha$, by $x = \psi_{0k}(\alpha, \beta)$ with $\psi_{0k}(\alpha, \beta) = \alpha(x_{i_{k2}} - x_{i_{k1}}) + \beta(x_{i_{k3}} - x_{i_{k1}}) + x_{i_{k1}}$.

Since the deformation is affine on the triangle, we have $\phi_k(x) = \psi_{1k}(\alpha, \beta)$ with $\psi_{1k}(\alpha, \beta) = \alpha(q_{i_{k2}}(1) - q_{i_{k1}}(1)) + \beta(q_{i_{k3}}(1) - q_{i_{k1}}(1)) + q_{i_{k1}}(1)$.

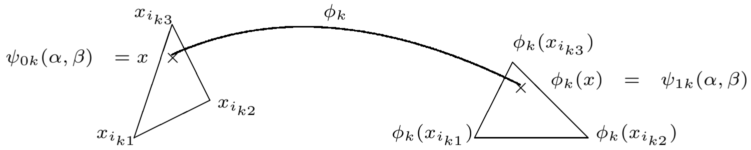

Fig. 3. Image of a point x in the template triangle T_k through the affine function ϕ_k

Using the coordinates (α, β) is in fact equivalent to making a new change of variable from the triangle T_k to the standard simplex $T_0 = \{\alpha + \beta < 1, \alpha, \beta > 0\}$ so that, denoting $A(T)$ for the area of a triangle T, and $s = (\alpha, \beta)$:

$$\int_{T_k} |I_1(\phi_k(x)) - I_0(x)|^2 |d_x\phi_k| dx = \int_{T_0} |I_1(\psi_{1k}(s)) - I_0(\psi_{0k}(s))|^2 A(T_k')ds \ . \quad (14)$$

This yields the final expression of the energy : $\mathcal{E}(p(0)) =$

$$\frac{1}{2}\langle K(q(0))p(0), p(0)\rangle + \lambda \sum_{k=1}^{r} \int_{T_0} |I_1(\psi_{1,k}(s)) - I_0(\psi_{0,k}(s))|^2 A(T_k')ds \ . \quad (15)$$

4 Computation of the Derivatives

4.1 Gradient

We compute the first derivative of g, which is needed for the gradient descent algorithm and the computation of the transversality equation. To compute this gradient we use formula (15) which can be differentiated without requiring Green's formula which would involve an integration over the edges of the triangles. We expect in particular more numerical accuracy from surface intergrals than from interpolated line integrals.

Proposition 1. *Denote* $z_k = (q_{k1}^1(1), q_{k2}^1(1), q_{k3}^1(1), q_{k1}^2(1), q_{k2}^2(1), q_{k3}^2(1)) \in \mathbb{R}^6$, *considered as a column vector and with a slight abuse of notation, denote* $A(z_k) = A(T_k')$. *Let* $z = (z_1, ..., z_N)^T$ *be the vector containing all the vertices of the triangles. We can notice that in* z, *some of the landmarks are repeated, but this* *does not affect the computation, since we treat each triangle separately. Let $\tilde{I}_{i,k} = I_i \circ \psi_{i,k}$ for $i = 0, 1$. The gradient of the data attachment term is equal to:*

$$\nabla g(z) = \sum_{k=1}^{r} \int_{T_0} \Big(2(\tilde{I}_{1,k}(s) - \tilde{I}_{0,k}(s)) A(z_k) \, (\partial_{z_k} \psi_{1,k}(s))^T \, \nabla I_1(\psi_{1,k}(s)) \\ + |\tilde{I}_{1,k}(s) - \tilde{I}_{0,k}(s)|^2 \, \nabla A(z_k)) ds \Big) \quad (16)$$

where $: \nabla A(z_k) = \begin{pmatrix} 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \end{pmatrix} z_k$ *and* $\partial_{z_k} \psi_{1,k} = \begin{pmatrix} 1-\alpha-\beta & \alpha & \beta & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-\alpha-\beta & \alpha & \beta \end{pmatrix}.$

4.2 Second Differential of *g*

We now compute the Hessian matrix of g which is needed for the implementation of Newtons's method.

Proposition 2. *Using the same notation as before, the second derivative of the data attachment term with respect to the final landmarks equals :*

$$d_z^2 g(\delta z, \delta z) = \sum_{k=1}^{r} \int_{T_0} (\delta z_k)^T \Big(2A(z_k) \, (\partial_{z_k} \psi_{1,k})^T \, \nabla I_1(\psi_{1,k}) \, \nabla I_1(\psi_{1,k})^T \, \partial_{z_k} \psi_{1,k} \\ + A(z_k)(\tilde{I}_{1,k} - \tilde{I}_{0,k}) \, (\partial_{z_k} \psi_k)^T \, Hess_{I_1}(\psi_{1,k}) \, \partial_{z_k} \psi_{1,k} \\ + 2(\tilde{I}_{1,k} - \tilde{I}_{0,k})(\partial_{z_k} \psi_{1,k})^T \nabla I_1(\psi_{1,k})(\nabla A(z_k))^T \\ + (\tilde{I}_{1,k} - \tilde{I}_{0,k})^2 Hess_A(z_k) \, ds \Big) \, \delta z_k \quad (17)$$

where $Hess_A(z_k) \equiv \begin{pmatrix} 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \end{pmatrix}$ *and* $Hess_f$ *denotes for the hessian matrix of* f.

Proof: We use the same notation as in the computation of the first derivative. We can notice that $\partial_{z_k} \psi_{1,k}$ is independent of z_k and $\nabla A(z_k)$ is linear on z_k, so that the second derivative of $\psi_{1,k}$ with respect to z_k is null and we easily get the expression of $\mathrm{Hess}_A(z_k)$ as the matrix involved in its gradient. This yields :

$$d_z^2 g(\delta z, \delta z) = \\ \sum_{k=1}^{r} \int_{T_0} \Big(2\langle \nabla I_1(\psi_{1,k}), \partial_{z_k} \psi_{1,k}(\delta z_k) \rangle \langle \nabla I_1(\psi_{1,k}), \partial_{z_k} \psi_{1,k}(\delta z_k) \rangle A(z_k) \\ + 2 d^2_{\psi_{1,k}} I_1(\partial_{z_k} \psi_{1,k}(\delta z_k), \partial_{z_k} \psi_{1,k}(\delta z_k))(I_1(\psi_{1,k}) - I_0(\psi_{0,k})) A(z_k) \\ + 2(I_1(\psi_{1,k}) - I_0(\psi_{0,k})) \langle \nabla I_1(\psi_{1,k}), \partial_{z_k} \psi_{1,k}(\delta z_k) \rangle \langle \nabla A(z_k), \delta z_k \rangle \\ + 2(I_1(\psi_{1,k}) - I_0(\psi_{0,k}))^2 \, d^2_{z_k} A(\delta z_k, \delta z_k) \Big) ds \ . \quad (18)$$

Equation (17) is the matrix form of (18).

5 Experiments and Discussion

In the experiments showed in figure 4, the first line corresponds to the final results of the gradient descent in the initial momentum space. The second line corresponds to the results of Newton's method. The deformation ϕ (fourth column) and the transformation of the template (third column) are computed using the interpolation formula ; it is the reconstructed diffeomorphism and no more its approximation by a piecewise affine function.

The mesh can be either adaptated to the template or be shared by every images. The choice depends on the goal we pursue. Using a common mesh enables a comparison of the resulting energies on the same area of the images (see table 1 and 2). In case of image detection or classification, we try to explain an image made of two different parts: a specific zone where the information is located and the background. If we want to give a probalistic model to each part, localizing the information, that is to say using an adaptative mesh, will probably enable to reach better results. The risk with object adapted triangulation is the data attachment term can be small when the deformed template is included in the target, but not perfectly aligned to it. This can happen in particular when the grey-level information is weak within the shape, espescially with binary images.

In each case, more iterations are needed by the gradient descent, often with less accurate results than with Newton's method.

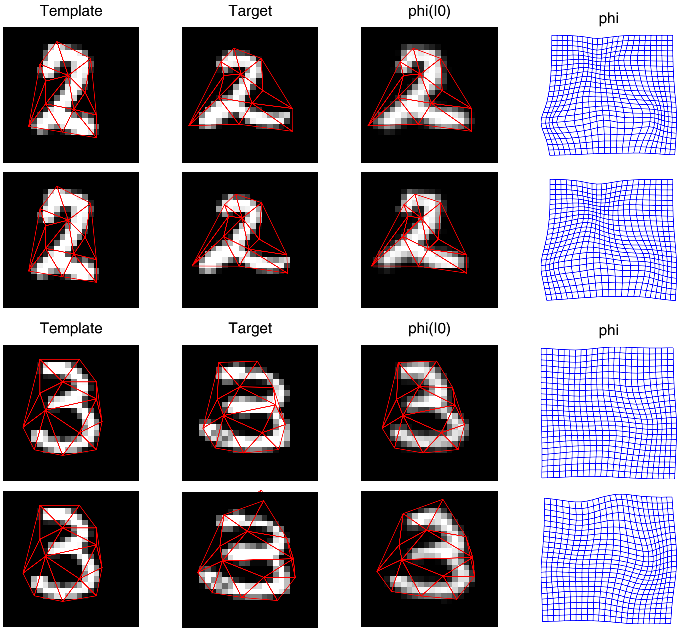

Fig. 4. Comparison between gradient descent (line 1 and 3) and root-finding (line 2 and 4) methods on an adaptative mesh for 2 different digits

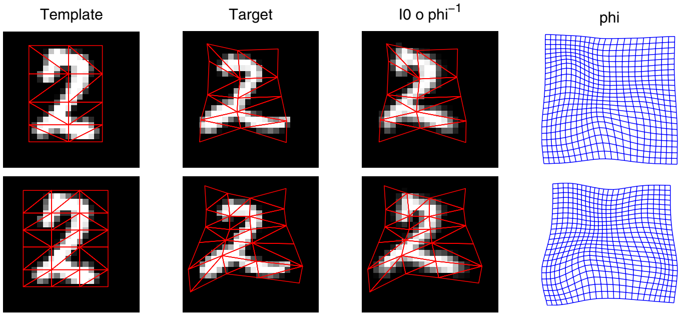

Fig. 5. Combination of gradient descent and root-finding methods for 2 regular mesh (15 and 24 landmarks)

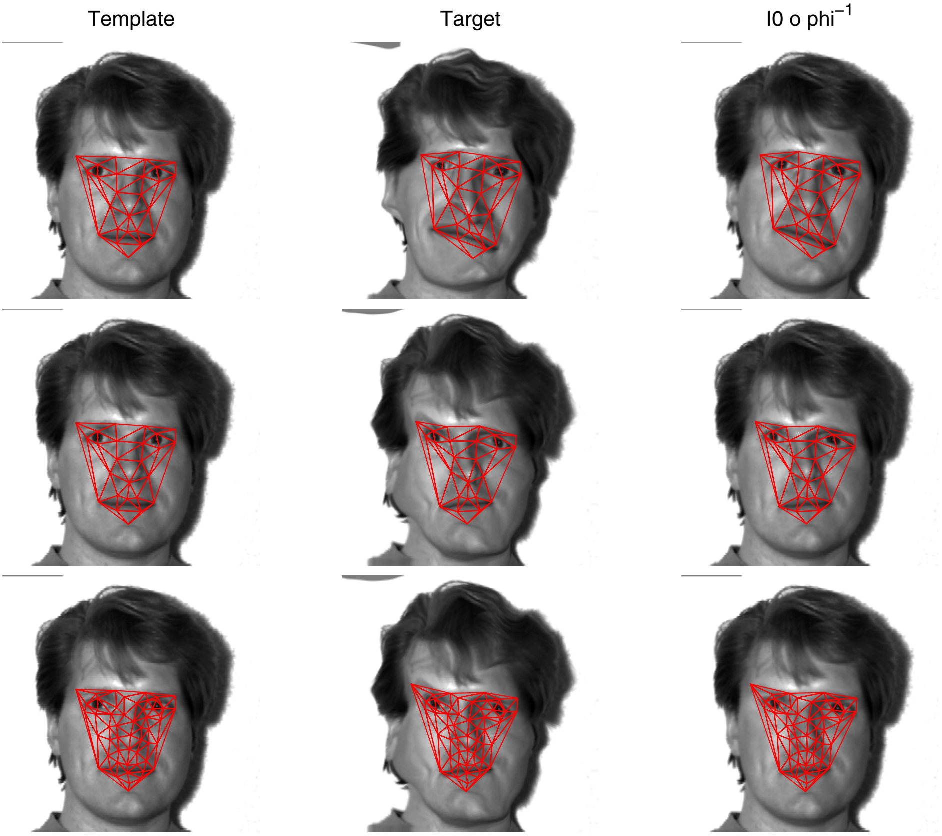

Fig. 6. Newton's method results on 2 synthetic face matchings (line 1 and 2), using 2 different meshes (line 2 and 3)

The number of singular directions used for Newton's method is computed automatically: we start with 10% of singular directions and keep adding new ones unless the norm of $G(p_0) = p_1(p_0) + \lambda \nabla g$ increases. The resulting energy is smaller using Newton's method as well as the averaged numerical value of $|G(p_0)|$. Typical initial values are larger than 300 for the energy and than 4 for the average of $|G(p_0)|$. Note that this value is not always 0 at the end of the iteration, essentially due to interpolation errors.

In figure 5, we can see the final results of the combination of both gradient and Newton's methods for a common regular mesh with 15 or 24 landmarks. If we increase the number of points, a good initialization of Newton's method is required. One solution is to combine the two methods as previously described. Handwritten digit images are almost binary, small images; this creates a risk of numerical unstability for the computation of their gradient and second derivative. For face images (100 times bigger), Newton's algorithm is more stable and uses almost every singular values in the last steps. The final result depends on the two parameters λ and σ_V. Increasing λ allows larger deformations to better fit the data, but the minimum is harder to achieve. The kernel parameter, σ_V, needs to be large enough to ensure triangle consistency, but small enough to avoid too rigid deformations (like in figure 6, 3^{rd} line). The tradeoff we made is choose σ_V almost equal to the size of the triangles. The design of the triangulation is important too. Indeed, since the deformation is affine on each triangle, all elements in one triangle will have a homogeneous displacement. Thus, it is reasonable to ensure that every triangle holds only one structure of the image, for example the mouth or the cheeks but not both.

6 Conclusion

We have presented here a new method for image matching using a triangulation of a restricted part of the image domain, and a piecewise affine transformation on this triangulation. We also introduced a new way for finding the transformation by directy solving the transversality equation. The motivation was to take advantage of the dimensionality reduction that is provided by the landmark dependence of the deformation and the linearity of the affine function that enables an explicit computation of the derivatives of the data attachement term. Solving the transversality equation by Newton's algorithm also provided significant acceleration of the convergence of our matching algorithm. A 3D generalization of the computations is also almost straightforward.

References

1. M. F. Beg, M. I. Miller, J. T. Ratnanather, A. Trouvé, L. Younes *Computing Metrics on Diffeomorphisms for Computational Anatomy*, International Journal of Computer Vision, (2004)
2. M. F. Beg, M. I. Miller, A. Trouvé, L. Younes, *Computing large deformation metric mappings via geodesic flow of diffeomorphisms*, Int. J. Comp. Vis., 61 (2005), pp 139-157

3. F.L. Bookstein *Morphometric tools for landmark data; geometry and biology*, Cambridge University Press, (1991)
4. R. Broit, C. Bajcsy, *Matching of deformed images*, Proc. 6th Int. Conf. of Pattern Recogition, München, (1982) pp 351-353
5. V. Camion, L. Younes *Geodesic Interpolating Splines*, In M. Figueiredo, J. Zerubia, and A.K. Jain, editors, Proceedings of EMMCVPR 2001, volume 2134 of Lecture Notes in Computer Science, Springuer (2001), pp 513-527
6. P. Dupuis, U. Grenander, M. I. Miller *Variational problems on flows of diffeomorphisms for image matching*, Quart. Appl. Math., vol. LVI, pp 587-600, (1998)
7. L. Garcin, A Rangarajan, L.Younes *Non rigid registration of shapes via diffeomorphic point matching and clustering*
8. J. Glaunes, A. Trouvé, L. Younes *Modeling planar shape variation via hamiltonian flow of curves* (Submitted)
9. J. Glaunes, M. Vaillant, M. I. Miller *Landmark matching via large deformation diffeomorphisms on the sphere*, J. Math. Imaging Vision 20 no. 1-2, (2004) pp 179-200.
10. U. Grenander, M.I. Miller *Computational anatomy: An emerging discipline*, Quaterly of Applied Mathematics, LVI (1998) pp 617-694
11. D.D. Holm, J.T. Ratnanather, A. Trouvé, L. Younes *Solition Dynamics in Computational Anatomy*, Neuroimage (2004)
12. D.G. Hull *Optimal Control Theory for Applications*, XIX, Sringer, (2003)
13. S. Joshi, M. I. Miller *Landmark matching via Large Deformation diffeomorphisms*, IEEE transaction in image processing 9 (2000) pp 1357-1370
14. S. Marsland, C Twining *Clamped-plate splines and the optimak flow bounded diffeomorphisms*, Complex Stochastic Systemp and Engineering, Oxford: Clarendon press (2002), pp 85-103.
15. M. I. Miller, A. Trouvé, L. Younes *On the metrics and Euler Lagrange Equations of Computational Anatomy*, Annual Review of Biomedical Engeneering 4 (2002) pp 375-405
16. M. I. Miller, L. Younes *Group action, diffeomorphism and matching: a general framework*, Int. J. Comp. Vis., 41, (2001) pp 61-84
17. F. Richard, L. Cohen *A new image registration technique with free boundary constraints : application to mammography*, Computer Vision and Image Understanding, vol. 89/2-3 pp 166-196, Special Issue on Nonrigid Registration, 2003
18. A. Trouvé *Diffeomorphisms Groups and Pattern Matching in Image Analysis*, International Journal of Computer Vision 28(3) (1998) pp 213-221
19. A. Trouvé, L. Younes *Local geometry of deformable template*, SIAM Journal of Numerical Analysis (2004-5)
20. M. Vaillant, J. Glaunes *Surface Matching via Currents*, Proceedings of Information Processing in Medical Imaging (IPMI 2005), pp. 381-392, 2005.
21. M. Vaillant, M. I. Miller, A. Trouvé, L. Younes *Statistics on diffeomorphisms via tangent space representaion*, NeuroImage 23 (2004) S161-169
22. G. Wahba *Spine Models for Observational Data*, Philadelphia, PA, SIAM, (1990)
23. L. Younes *Optimal matching between shapes via elastic deformations*, Image and Vision Computing, (1999)
24. L. Younes *Computable elastic distances between shapes*, SIAM J Appl Math, 58 (1998) pp 565-586

Appendix

* We provide here for completness, a sketch of the derivation of the Hamiltonian formulation given in (4) and of the transversality condition (7).
  Let $(v_t, q(t))_{t\in[0,1]}$ be a minimizer of (2) with $v \in L^2([0,1], V)$. For any perturbation $v_t \to v_t + \epsilon h_t$ with $h \in L^2([0,1], V)$, we get at $\epsilon = 0$

$$\partial_\epsilon \dot{q}(t) = d_{q(t)} v_t \partial_\epsilon q(t) + h_t(q(t)) \tag{19}$$

where $h_t(q(t)) \doteq (h_t(q_i(t)))_{1\leq i\leq N}$. Let $(P_{s,t})$ be the matrix semi-group satisfying

$$P_{s,s} = \mathrm{Id}_{nN} \text{ and } \partial_t P_{s,t} = d_{q(t)} v_t P_{s,t},\ \forall t \geq s\,. \tag{20}$$

From (19) and (20), we get at $\epsilon = 0$, $\partial_\epsilon q(1) = \int_0^1 P_{s,1} h_s(q(s)) ds$ and

$$\partial_\epsilon E(v, q(1)) = \int_0^1 \langle v_s, h_s\rangle_V ds + \int_0^1 \langle \nabla_{q(1)} g, P_{s,1} h_s(q(s))\rangle_{\mathbb{R}^{nN}} ds = 0.$$

Since h is arbitrary, we get $v_s(q(s)) = \dot{q}(s) = K(q(s))p(s) = \frac{\partial H}{\partial p}(q(s), p(s))$ where $p(s) + P^*_{s,1}\nabla_{q(1)} g = 0$ which gives the first equation of (4) and also (7) for $s = 1$. From (20), we get $\partial_s P_{s,t} = -P_{s,t} d_{q(s)} v_s$ so that eventually

$$\dot{p}(s) = \partial_s P^*_{s,1} p_1 = -(d_{q(s)} v_s)^* p(s) = -\frac{\partial H}{\partial q}(q(s), p(s))\,. \tag{21}$$

* We provide here a proposition concerning the triangle consistency.

Table 1. Comparison of the 2 metods for solving the Image matching problem for handwritten digits (images normalized in $[-1, 1]$)

	Energy value		Mean value $\lvert G(p(0))\rvert$	
Fig	Gradient desc.	Newton's method	Gradient desc.	Newton's method
Fig 4 1st line	62.87	60.43	0.95	0.48
Fig 4 2nd line	166	156	1.30	0.62
Fig 5 15 pts	107	76.9	0.76	0.33
Fig 5 24 pts	71.1	65.5	0.58	0.40

Table 2. Newton's method results on face images (images normalized in $\{0,\ldots,255\}$)

Fig	Energy value	Mean value of the $\lvert G(p(0))\rvert$ vector
Fig 6 1st line	3.3910^3	1.08
Fig 6 2nd line	$1.98.10^3$	0.40
Fig 6 3rd line	$1.32.10^3$	0.09

Proposition 3. *Let $\gamma(t) = \sin(\theta(t))$ where $\theta(t)$ is one of the triangle angles. Let V be a self reproducing kernel Hilbert space, with a σ^2 variance gaussian kernel and ϕ_t be the diffeomorphism solution of $d\phi/dt = v_t \circ \phi_t$ for a velocity vector field $v_t \in L^1([0,1],V)$. Denoting $\psi(x) = 2xe^{2x}$, a sufficient condition to keep the triangle consistency is given by*

$$\psi(\frac{\sqrt{2\lambda g_{I_1,I_0}(q(0))}}{\sigma}) \leq \frac{|\gamma(0)|}{(1+|\gamma(0)|)}\,.$$

Proof: Let A, B, C be the 3 vertices of a triangle, $a(t) = \phi_t(B) - \phi_t(A)$ and $b(t) = \phi_t(C) - \phi_t(A)$. We want to control the sign of the sine of the $\widehat{BAC}$ angle, θ_t. To avoid reversal of the triangle this quantity must not change its sign. Let $\alpha(t) = |a(t)\ b(t)| = |a(t) \wedge b(t)|$; we can notice that : $\alpha(t) = |a(t)||b(t)|\sin(\theta_t)$. Then, using Cauchy-Schwarz inequality:

$$\begin{aligned}\partial_t\alpha(t) = \langle \partial_t a(t) \wedge b(t) + a(t) \wedge \partial_t b(t), \frac{a(t)\wedge b(t)}{|a(t)\wedge b(t)|}\rangle \\ \leq (|\partial_t a(t)||b(t)| + |\partial_t b(t)||a(t)|)\ .\end{aligned}$$

But, $\partial_t a(t) = \partial_t(\phi_t(B) - \phi_t(A)) = v_t(\phi_t(B)) - v_t(\phi_t(A))$. So that:

$$\partial_t\alpha(t) \leq 2\|dv_t\|_\infty |a(t)||b(t)|\ . \tag{22}$$

Let $\gamma(t) = \sin\theta_t = \frac{\alpha(t)}{|a(t)||b(t)|}$; we try to quantify the difference between $\sin(\theta_t)$ and $\sin(\theta_0)$ to find a suffitient condition.

$$\begin{aligned}&\partial_t\gamma(t)\\ &= \frac{\partial_t\alpha(t)}{|a(t)||b(t)|} - \frac{\alpha(t)}{|a(t)|^2|b(t)|^2}(|b(t)|\langle\partial_t a(t), \frac{a(t)}{|a(t)|}\rangle + |a(t)|\langle\partial_t b(t), \frac{b(t)}{|b(t)|}\rangle)\\ &\leq \frac{1}{|a(t)||b(t)|}\left(|\partial_t\alpha(t)| + |\alpha(t)|\left(\left|\langle\frac{\partial_t a(t)}{|a(t)|}, \frac{a(t)}{|a(t)|}\rangle\right| + \left|\langle\frac{\partial_t b(t)}{|b(t)|}, \frac{b(t)}{|b(t)|}\rangle\right|\right)\right)\ .\end{aligned}$$

Using (22), $\partial_t\gamma(t) \leq 2\|dv_t\|_\infty + |\gamma(t)|\left(\left|\frac{\partial_t a(t)}{a(t)}\right| + \left|\frac{\partial_t b(t)}{b(t)}\right|\right) \leq 2\|dv_t\|_\infty(1 + |\gamma(t)|)$. And $|\gamma(0) - \gamma(t)| \leq \int_0^t |\partial_t\gamma(t)|dt \leq \int_0^t 2\|dv_t\|_\infty(1 + |\gamma(0)|)dt + \int_0^t 2\|dv_t\|_\infty|\gamma(0) - \gamma(t)|dt$. Applying Gronwall's lemma to this last inequality, we finally get:

$$|\gamma(0) - \gamma(t)| \leq 2(1 + |\gamma(0)|)\left(\int_0^1 \|dv_t\|_\infty dt\right)\exp\left(2\int_0^1 \|dv_t\|_\infty dt\right).$$

As we are using a self reproducing gaussian kernel Hilbert space: $\forall x \in \mathbb{R}^d$ $|v(x)| = \sup_{|a|\leq 1}\langle v(x), a\rangle_{\mathbb{R}^d} = \sup_{|a|\leq 1}\langle K_x a, v\rangle_V$, so: $\|v\|_\infty \leq |||K_{x,x}|||\|v\|_V = \|v\|_V$, where $|||K_{x,x}|||$ is the matrix norm subordinate to the Euclidian norm in $\mathbb{R}^d$, and, using a Taylor development of the kernel, $\|dv_t\|_\infty \leq \frac{1}{\sigma}\|v_t\|_V$. So we get: $\int_0^1 \|dv_t\|_\infty dt \leq \frac{1}{\sigma}\sqrt{2E_k(v)} \leq \frac{\sqrt{\lambda}}{\sigma}\tilde{G}$ where $\tilde{G} \doteq \sqrt{2g_{I_1,I_0}(q(0))}$. And finally: $\forall v \in L^1([0,1],V)$,

$$|\gamma(0)-\gamma(t)| \leq (1+|\gamma(0)|)\psi(\frac{\sqrt{\lambda}}{\sigma}\tilde{G}) \text{ where } \psi(x) = 2xe^{2x},\ \forall x \geq 0\,. \tag{23}$$

To avoid the reversal of a triangle, it suffices that $|\gamma(0)-\gamma(t)| \leq |\gamma(0)|$ for any $t \in [0,1]$. A sufficient condition is $\psi(\frac{\sqrt{\lambda}}{\sigma}\tilde{G}) \leq \left(\frac{|\gamma(0)|}{(1+|\gamma(0)|)}\right)$, which gives the result.