Bidirectional Composition on Lie Groups for Gradient-based Image Alignment

Rémi Megret, Jean-Baptiste Authesserre, Yannick Berthoumieu

Bidirectional Composition on Lie Groups for Gradient-based Image Alignment

Rémi Mégret, Jean-Baptiste Authesserre* and Yannick Berthoumieu

***Abstract*—In this paper, a new formulation based on Bidirectional Composition on Lie Groups (BCL) for parametric gradient-based image alignment is presented. Contrary to the conventional approaches, the BCL method takes advantage of the gradients of both template and current image without combining them a priori. Based on this bidirectional formulation, two methods are proposed and their relationship with state-of-the-art gradient based approaches is fully discussed. The first one, i.e. the BCL method, relies on the compositional framework to provide the minimization of the compensated error with respect to an augmented parameter vector. The second one, the Projected BCL (PBCL), corresponds to a close approximation of the BCL approach. A comparative study is carried out dealing with computational complexity, convergence rate and frequence of convergence. Numerical experiments using a conventional benchmark show the performance improvement especially for asymmetric levels of noise, which is also discussed from a theoretical point of view.**

***Index Terms*—Bidirectional image alignment, image registration, gradient methods, Lie groups.**

I. Introduction

Image alignment is a fundamental task of many vision applications. Over the last decades, numerous works proposed various approaches to solve registration adapted to conventional application fields, i.e. object tracking, image mosaicking, video compression or augmented reality using day light video. For applications such as radar imaging, night vision enhancement in road and air traffic, thermal imaging and medical electronics, image alignment is still a challenging issue leading to register sets of images which have very low Signal to Noise Ratio (SNR) because of their typical signal-degradation (e.g. photon, electronics, speckle and quantization). For instance, a night vision system provides an alternative means of improving visibility under low or no-light conditions. Because the light reflected by a target is very weak, the image captured by a Low-Light-Level camera has a great deal of noise attached on it, i.e. shot noise. In such a low-SNR framework, computer vision tasks including image filtering, super-resolution imaging, segmentation, or recognition require alignment of images which are characterized by different quality, i.e. filtered image versus very noisy current image for instance. In order to provide tractable approaches to work with various contexts of symmetric or asymmetric levels of noise, one has to consider the appropriateness of the competing image alignement algorithms.

Image alignment based on template matching is a natural approach to image registration, by estimating the parameters that best warp one image onto the other. The estimation is conventionally provided by the minimization of the displaced frame difference between the template and the current image. Since the Lucas and Kanade Forwards Additive algorithm [1], most approaches have been formulated using such a unidirectional compensation approach. The two most natural approaches (Forwards and Inverse) have been discussed thoroughly by Baker and Matthews [2]. The authors proposed exhaustive experimental and numerical investigations of four main classes: Forwards Additive, Forwards Compositional, Inverse Additive, and Inverse Compositional. The approaches were extended to take into account asymmetric levels of spatial resolution in [3].

In [4] and [5], Benhimane et al. proposed a novel optimization strategy, the Efficient Second-order Minimization (ESM). They introduced the Lie Group parameterization of motion and the use of the gradients of both image and template yielding the elimination of the second-order terms of the error. The ESM algorithm achieves better convergence and robustness properties for only a slight overhead compared to the more standard Gauss-Newton approach [2].

Structurally, the methods considered above assume a fully asymmetric or symmetric solution to the alignment issue, i.e. using either template or current image gradients [2] [3], or using both in a symmetric manner [4] [5]. However, in some real applications, noise may have an intermediate level of asymmetry, corrupting differently template and current image, which leads to the suboptimality of a fully asymmetric or symmetric assumption. In this context, our proposal consists in finding an alternative way to solve the alignment issue which allows the algorithm to adapt to asymmetric levels of noise. To do this, a generic derivation of image alignment based on template matching is provided.

Our contribution presented in this paper builds on the bidirectional composition framework which has been briefly introduced in [6]. In this contribution, we:

- propose an original formulation based on Bidirectional Composition on Lie Groups and a related BCL algorithm,
- provide an alternate interpretation of the BCL approach using a novel Projected BCL (PBCL) algorithm, which is shown to be equivalent to the BCL, and have second-order minimization properties thanks to its connection with the ESM algorithm [5].
- show that the proposed generic approaches yields more

Manuscript received July 27, 2009. The authors are with the Signal and Image Processing Group, IMS laboratory, University of Bordeaux, 351 Cours de la Libration F-33405 TALENCE CEDEX, France, (e-mail: remi.megret@ims-bordeaux.fr, jean-baptiste.authesserre@ims-bordeaux.fr, yannick.berthoumieu@ims-bordeaux.fr)

EDICS: TEC-ISR

robust algorithms than the generic state of the art algorithms on a larger variety of relative noise levels, which is supported by both a theoretical discussion and experimental evaluations, and draw recommendations on the situations in which to use each approach.

This paper is structurated in four parts. In section II, the background in parametric image alignment is presented. In section III, the image alignment problem is formalized within the bidirectional composition framework, from which a comprehensive set of Lie Group approaches is introduced. The novel Bidirectional Compositional Lie algorithm is discussed in section IV, where its theoretical relationship and gain with respect to existing methods is studied. The performances of the approaches are then presented in section V.

II. Background

A. Conventional image alignment approach

Image alignment between an image I and an image template T is conventionally expressed [1] as minimizing an error $E(\boldsymbol{\mu}_I)$ between the two images after warping I forward onto T:

$$\bar{\boldsymbol{\mu}}_I = \underset{\boldsymbol{\mu}_I}{\operatorname{argmin}}\ E(\boldsymbol{\mu}_I), \text{ with} \quad (1)$$

$$E(\boldsymbol{\mu}_I) = \sum_{\mathbf{x}_i \in R} \left| I(\mathbf{W}(\boldsymbol{\mu}_I, \mathbf{x}_i)) - T(\mathbf{x}_i) \right|^2, \quad (2)$$

where:

- The motion model is represented by a warp function $\mathbf{W}(\boldsymbol{\mu}, \mathbf{x})$ of parameter $\boldsymbol{\mu} \in \mathcal{P}$, operating at $\mathbf{x}$.
- $R = (x_1, \ldots, x_N)$ is a discrete sampling of the region of interest in the template coordinate frame.

The error function E is then minimized using a gradient based optimization technique [2]. Each iteration n is based on an incremental parameterization from an initial estimate $\boldsymbol{\mu}_I^n$. This may be an additive increment, such as in the original Forwards Additive (FA) approach [1]:

$$\boldsymbol{\mu}_I = \boldsymbol{\mu}_I^n + \delta\boldsymbol{\mu}, \quad (3)$$

or a compositional increment, as in the Forwards Compositional (FC) approach [7]:

$$\boldsymbol{\mu}_I = \boldsymbol{\mu}_I^n \circ \delta\boldsymbol{\mu}. \quad (4)$$

As it was pointed out in [8], it is more natural to use a compositional incremental parameterization for spatial transformation because of its geometric meaning, which will be the approach used in the sequel.

In order to be able to describe the framework in a simple and consistent way, we require that the considered motion model has group action properties. The parameter space $\mathcal{P}$ forms a group, which acts on image coordinates $\mathbf{x}$ through $\mathbf{W}$. This action has the following properties, which are related respectively to composition ($\circ$), inversion ($^{-1}$) and parameters of the identity transformation 0:

$$\mathbf{W}(\boldsymbol{\mu} \circ \delta\boldsymbol{\mu}, \mathbf{x}) = \mathbf{W}(\boldsymbol{\mu}, \mathbf{W}(\delta\boldsymbol{\mu}, \mathbf{x})), \quad (5)$$

$$\mathbf{y} = \mathbf{W}(\boldsymbol{\mu}^{-1}, \mathbf{x}) \Leftrightarrow \mathbf{x} = \mathbf{W}(\boldsymbol{\mu}, \mathbf{y}), \quad (6)$$

$$\mathbf{W}(0, \mathbf{x}) = \mathbf{x}. \quad (7)$$

These group action properties are sometimes refered in the litterature as group of transformations. Parametric motion models such as affine or homography motion satisfy those properties [5]. Non rigid models satisfying the group action properties have been discussed in [8].

B. Alternative approaches

In previous algorithms, the optimization is based on the gradients of image I only. The inverse compositionnal (IC) algorithm proposed in [9] instead considers warping the template T onto the image I, which corresponds to the minimization of the following error at step n:

$$E^n(\delta\boldsymbol{\mu}_T) = \sum_{\mathbf{x}_i \in R} \left| I(\mathbf{W}(\boldsymbol{\mu}_I^n, \mathbf{x}_i)) - T(\mathbf{W}(\delta\boldsymbol{\mu}_T, \mathbf{x}_i)) \right|^2, \quad (8)$$

where the estimate $\boldsymbol{\mu}_I^n$ is updated after each iteration according to the following rule:

$$\boldsymbol{\mu}_I^{n+1} \leftarrow \boldsymbol{\mu}_I^n \circ (\delta\boldsymbol{\mu}_T)^{-1}. \quad (9)$$

This approach uses the gradients of the template T only. Since they can be precomputed, this yields faster computation of each iteration.

The Efficient Second-order Minimization (ESM) algorithm [4] [5] is expressed as a forwards compositional approach based on (2) and (4), but uses additive combination of the gradients of both I and T within the optimization. The authors demonstrated faster convergence rates and better robustness, and proved theoretically the better convergence properties when both images are identical up to a compensation of true unknown parameters $\bar{\boldsymbol{\mu}}_I$:

$$\forall \mathbf{x} \in R \quad I(\mathbf{W}(\bar{\boldsymbol{\mu}}_I, \mathbf{x})) = T(\mathbf{x}). \quad (10)$$

This method will be discussed in more detail in section III-E.

The use of the gradients of both I and T was generalized in [10] into a weighted combination of the gradients. It was shown that this could improve the robustness in some situations, but the automatic computation of the optimal weights still requires manual tuning.

The bidirectional formulation of the error is introduced in [11], as Bidirectionnal Gradient Method (BDGM). It corresponds to the minimization of an error E computed after compensating both I and T. The authors propose a theoretical convergence analysis in both the far range phase (linear convergence) and near optimum phase (quadratic convergence) assuming the equality of the two images up to motion compensation (10). They showed the superiority of the BDGM algorithm over the unidirectional Forwards additive algorithm, by providing bounds on the decrease of the error at each iteration. However their approach uses an additive incremental parameterization and an update rule that was shown [6] to be unreliable in the non translational case, for example when both translations and rotations are combined.

In this paper the bidirectional formulation will be used and extended to a compositional update on Lie Groups, which will allow us to derive new approaches dealing naturally with asymmetricaly distributed noise.

C. Lie groups parameterization of motion

A common way to parameterize motion does not take into account any group properties and embeds the parameter space $\mathcal{P}$ into a vector space $\mathbb{R}^m$ of finite dimensions [2], [1], [7]. However when $\mathcal{P}$ is not a vector subspace the constraints that the parameters $\boldsymbol{\mu}$ have to remain in $\mathcal{P}$ have to be enforced explicitly, leading to the use of constrained optimization techniques [8].

A compositional Lie group $\mathcal{P}$ is a differentiable manifold structured by the compositional operation ($\circ$). This is the case of rigid models such as non degenerate affine motion for the euclidian plane $\mathbf{x} \in \mathbb{R}^2$ and homography for the projective plane $\mathbf{x} \in \mathbb{P}^2$ [12]. Using Lie group properties, instead of a vector space embedding, ensures that the solution belongs to $\mathcal{P}$ allowing the use of a simpler unconstrained optimization procedure.

A Lie algebra $\mathfrak{P}$ can be associated to a finite dimensional Lie Group $\mathcal{P}$ whose underlying finite dimensional vector space is the tangent space to $\mathcal{P}$ at the neutral element 0. The main idea behind Lie Group is that locally an increment $\delta\boldsymbol{\mu} \in \mathcal{P}$ around 0 can be bijectively reparameterized by an increment $\delta\mathbf{v} \in \mathfrak{P}$ using the exponental map:

$$\delta\boldsymbol{\mu}(\delta\mathbf{v}) = \exp(\delta\mathbf{v}), \tag{11}$$

with the following properties, for any $\alpha, \beta \in \mathbb{R}$:

$$\exp(-\delta\mathbf{v}) = \exp(\delta\mathbf{v})^{-1} \tag{12}$$

$$\exp(\alpha\delta\mathbf{v}) \circ \exp(\beta\delta\mathbf{v}) = exp((\alpha+\beta)\delta\mathbf{v}). \tag{13}$$

The term *Composition on Lie Groups*, abbreviated as Compositional Lie (CL) in the sequel, has been chosen to emphasize that the group action $\mathbf{W}$ is related to the composition of transformations, in contrast to additive parameterization, that have been used for dense motion fields [8] and for which relation (5) does not hold. Although vector space embedding or non group action versions of the proposed algorithms could also be derived, they will not be presented as they would not bring new insights in the scope of this study. The methodology of Brooks and Arbel [13] may be used to extend our results to such approaches.

The transformations which will be used in the experiments will be based on Lie group homography parameterization as presented in [5] or [12].

III. Bidirectional alignment on Lie groups

A. Problem formalization

Aligning two images can be formalized in a bidirectional way as estimating parameters $(\boldsymbol{\mu}_I, \boldsymbol{\mu}_T)$ such that the compensated images are most similar according to an image dissimilarity criterion. For pixel-wise L_2 objective function, the error can be defined as:

$$E(\boldsymbol{\mu}_I, \boldsymbol{\mu}_T) = \sum_{i \in 1..N} \left| e_i(\boldsymbol{\mu}_I, \boldsymbol{\mu}_T) \right|^2, \quad \text{with} \tag{14}$$

$$e_i(\boldsymbol{\mu}_I, \boldsymbol{\mu}_T) = I(\mathbf{W}(\boldsymbol{\mu}_I, \mathbf{x}_i)) - T(\mathbf{W}(\boldsymbol{\mu}_T, \mathbf{x}_i)). \tag{15}$$

The error is computed at each pixel $\mathbf{x}_i$ belonging to a region of interest $R = (\mathbf{x}_1, \dots \mathbf{x}_N)$ to form an error vector $\mathbf{e} = (e_i)_{i=1..N}$.

In the case of gradient based optimization, this error is minimized in an iterative scheme. At iteration n, the problem is therefore reparameterized around the initial parameters $(\boldsymbol{\mu}_I^n, \boldsymbol{\mu}_T^n) \in \mathcal{P}$ by using an incremental update vector $\delta\mathbf{v} \in \mathfrak{P}$:

$$E^n(\delta\mathbf{v}) = E(\boldsymbol{\mu}_I(\boldsymbol{\mu}_I^n, \delta\mathbf{v}), \boldsymbol{\mu}_T(\boldsymbol{\mu}_T^n, \delta\mathbf{v})), \tag{16}$$

where $\boldsymbol{\mu}_I(\boldsymbol{\mu}_I^n, 0) = \boldsymbol{\mu}_I^n$ and $\boldsymbol{\mu}_T(\boldsymbol{\mu}_T^n, 0) = \boldsymbol{\mu}_T^n$. The choice of the local reparameterization functions $\boldsymbol{\mu}_I(\boldsymbol{\mu}_I^n, \cdot)$ and $\boldsymbol{\mu}_T(\boldsymbol{\mu}_T^n, \cdot)$ has a strong influence on the convergence properties of the alignment procedure, as it will enforce restrictions in the exploration of the bidirectionnal parameter space. This choice will be discussed in subsection III-B. The derivation of the corresponding gradient based optimization procedure at the iteration level will be presented in subsection III-C.

In applications such as tracking, the region of interest R is defined specifically on the template image. In order to take into account clutter in the alignment procedure, one should avoid the drift of the region of interest by constraining $\boldsymbol{\mu}_T$ to stay close to the initial parameters $\boldsymbol{\mu}_T^0$. Once an estimate $\delta\hat{\mathbf{v}}$ is obtained at iteration n, the bidirectional parameters for next iteration are therefore obtained using an update rule:

$$(\boldsymbol{\mu}_I^{n+1}, \boldsymbol{\mu}_T^{n+1}) \leftarrow \text{update}(\boldsymbol{\mu}_I(\boldsymbol{\mu}_I^n, \delta\hat{\mathbf{v}}), \boldsymbol{\mu}_T(\boldsymbol{\mu}_T^n, \delta\hat{\mathbf{v}})). \tag{17}$$

The choice of an update rule and related convergence issues will be discussed in subsection III-D

Within this framework, the considered iterative algorithms can be summarized in a generic way as follows:

1. Define the reference T.
2. Define the current image I.
3. At iteration $n = 0$, initialize the bidirectional parameters $(\boldsymbol{\mu}_I^n, \boldsymbol{\mu}_T^n)$.
4. Estimate an optimum $\delta\hat{\mathbf{v}}$ of the locally reparameterized error $E^n(\delta\mathbf{v})$.
5. Apply the update rule (17) to obtain $(\boldsymbol{\mu}_I^{n+1}, \boldsymbol{\mu}_T^{n+1})$.
6. If not converged $n \leftarrow n+1$ and go to step 4.

B. Local reparameterization

Four main local reparameterization will be discussed: Forwards, Inverse, Symmetric and Bidirectional. The various parameters that are used in this formulation are illustrated in figure 2. Note that when using a Forwards update rule (see subsection III-D), $\boldsymbol{\mu}_T^n$ is always reinitialized to $\boldsymbol{\mu}_T^0$.

- **Forwards Compositional Lie (**FCL**).** The image is warped onto the template depending on the increment $\delta\mathbf{v} \in \mathfrak{P}$:

$$\boldsymbol{\mu}_I = \boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v}) \quad \text{and} \quad \boldsymbol{\mu}_T = \boldsymbol{\mu}_T^0. \tag{18}$$

- **Inverse Compositional Lie (**ICL**).** The template is warped onto the image depending on the increment $\delta\mathbf{v}$:

$$\boldsymbol{\mu}_I = \boldsymbol{\mu}_I^n \quad \text{and} \quad \boldsymbol{\mu}_T = \boldsymbol{\mu}_T^n \circ \exp(-\delta\mathbf{v}). \tag{19}$$

- **Symmetric Compositional Lie (**SCL**).** I and T are compensated towards each other symmetrically, with respect to a single parameter vector $\delta\mathbf{v}$:

$$\boldsymbol{\mu}_I = \boldsymbol{\mu}_I^n \circ \exp(\frac{1}{2}\delta\mathbf{v}) \text{ and } \boldsymbol{\mu}_T = \boldsymbol{\mu}_T^n \circ \exp(-\frac{1}{2}\delta\mathbf{v}). \tag{20}$$

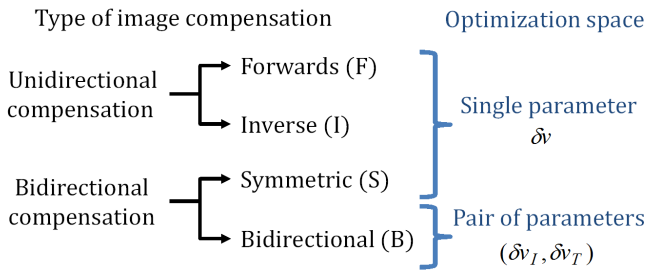

Fig. 1. Taxonomy of the FCL, ICL, SCL, BCL approaches showing two complementary ways of categorizing the discussed methods.

- **Bidirectional Compositional Lie (**BCL**).** The image I and the template T are both warped, using two independent sets of parameters $\delta\mathbf{v}_I$ and $\delta\mathbf{v}_T$. The increment vector is therefore bidirectional and can be decomposed as $\delta\mathbf{v} = (\delta\mathbf{v}_I, \delta\mathbf{v}_T) \in \mathfrak{P}^2$, where each component parameterizes the compensation of only one of the image:

$$\boldsymbol{\mu}_I = \boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v}_I) \text{ and } \boldsymbol{\mu}_T = \boldsymbol{\mu}_T^n \circ \exp(-\delta\mathbf{v}_T). \quad (21)$$

We showed in [6] how to integrate existing approaches within such a framework, but focused on vector space parameter embedding such as additive ($\boldsymbol{\mu} = \boldsymbol{\mu}^n + \delta\boldsymbol{\mu}$) and compositional ($\boldsymbol{\mu} = \boldsymbol{\mu}^n \circ \delta\boldsymbol{\mu}$) local reparameterization. We will consider in the sequel the corresponding extensions to a Lie Group parameterization. The FCL and ICL approaches extend the Forwards Compositional [7] and Inverse Compositional approaches [9]. The SCL and BCL approaches correct and extend the symmetric and the bidirectional approaches formulated in an additive manner in [11]. The SCL is also related to the ESM algorithm [5], which we will discuss in detail. This will allow us to unify the formulation on a single kind of parameterization, and facilitate the comparative analysis of the new BCL algorithm with other approaches.

The local reparameterizations can be classified based on the nature of the increment vector $\delta\mathbf{v}$. This increment can be considered to be homogeneous to a single parameter vector for the FCL, ICL and SCL approaches or to a pair of parameter vectors $(\delta\mathbf{v}_I, \delta\mathbf{v}_T)$ for the BCL approach.

Alternatively, a second classification instead considers which image is affected by the motion compensation with local parameter $\delta\mathbf{v}$. The Forwards and Inverse approaches consider unidirectional compensation (one image is fixed, while the other is moving). The Symmetric and Bidirectional approaches consider bidirectional compensation (the image and the template are both moving).

Those two complementary classifications are summed up in Figure 1. We note that the SCL approach uses a unidirectional local reparameterization, but considers the compensation of both images. We will show in section III-E how this approach relates to the ESM algorithm [4] [5].

The BCL approach uses a bidirectional local reparameterization $(\delta\mathbf{v}_I, \delta\mathbf{v}_T)$. This property makes it very particular compared to the other approaches, and which explains its novel properties. This will be discussed in section IV.

C. Optimization technique

Any optimization algorithm could be used in step 4. We will develop the framework with a Gauss-Newton (GN) estimation, based on the conclusions of the detailed comparative study that Baker and Matthews have done in the context of image alignment [2]. GN provides fast convergence rate and high convergence frequency for a reasonnable computational cost when compared to other second-order optimization techniques. This approach asssumes that the warp function $\mathbf{W}$ is differentiable w.r.t both $\delta\mathbf{v}$ and $\mathbf{x}$ and that the composition and inverse maps are also differentiable w.r.t their arguments.

The GN optimization of the generic error function (16) is based on the linearization of the vectorial error $\mathbf{e}$:

$$\mathbf{e}(\delta\mathbf{v}) = \mathbf{e}(0) + J(0)\delta\mathbf{v} + o(\|\delta\mathbf{v}\|), \quad (22)$$

where $J(0)$ corresponds to the Jacobian matrix of the error vector $\mathbf{e}$ around the initialization. For the sake of notational simplification, the point of evaluation will be omitted when it corresponds to $\delta\mathbf{v} = 0$ if there is no ambiguity.

$$J(\delta\mathbf{v}) = \left.\frac{\partial\mathbf{e}(\mathbf{v})}{\partial\mathbf{v}}\right|_{\delta\mathbf{v}} \quad (23)$$

This matrix can be expressed as the concatenation $J(0) = [{J_1}^t(0), \dots {J_N}^t(0)]^t$ of the gradients $J_i(0)$ of the pixelwise errors $\mathbf{e}_i$. This yields the generic solution:

$$\delta\mathbf{v} = -(J(0))^+ \ \mathbf{e}(0) = -(J^t(0)J(0))^{-1}(J^t(0)\mathbf{e}(0)), \quad (24)$$

where $(J(0))^+$ stands for the pseudo-inverse. In practice, the pseudo-inverse can be precomputed when $J(0)$ remains fixed accross iterations (ICL approach). Else, it may be more efficient to use the second form shown, which was proposed in the seminal work of Lucas and Kanade algorithm [1] [2]. This equation is over-constrained for a full rank matrix $J(0)$, with more rows than columns. This is most of the time the case for rigid image alignment if the number N of pixels is larger than the number p of parameters.

For the degenerate cases stemming from a too large number of parameters to estimate, for instance when estimating a dense displacement field, special care has to be taken into account by introducing additional regularization terms to avoid degeneracy. These extensions of the error formulation and the optimization procedure will not be detailed in the sequel, and we refer the reader to [14] where a classification of the possible regularization approaches is proposed.

The Jacobian matrix J is specific for each approach. In the case of Lie group parameterization (18), (19), (20) and (21), all of them can be expressed using the Jacobian matrices J_I^{CL} and J_T^{CL} of the image I and T with respect to the incremental compositional Lie algebra parameter $\delta\mathbf{v}$:

$$J_{I,i}^{CL}(\delta\mathbf{v}) = \left.\frac{\partial I(\mathbf{W}(\boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v}), \mathbf{x}_i))}{\partial\delta\mathbf{v}}\right|_{\delta\mathbf{v}} \quad (25)$$

$$J_{T,i}^{CL}(\delta\mathbf{v}) = \left.\frac{\partial T(\mathbf{W}(\boldsymbol{\mu}_T^n \circ \exp(\delta\mathbf{v}), \mathbf{x}_i))}{\partial\delta\mathbf{v}}\right|_{\delta\mathbf{v}} \quad (26)$$

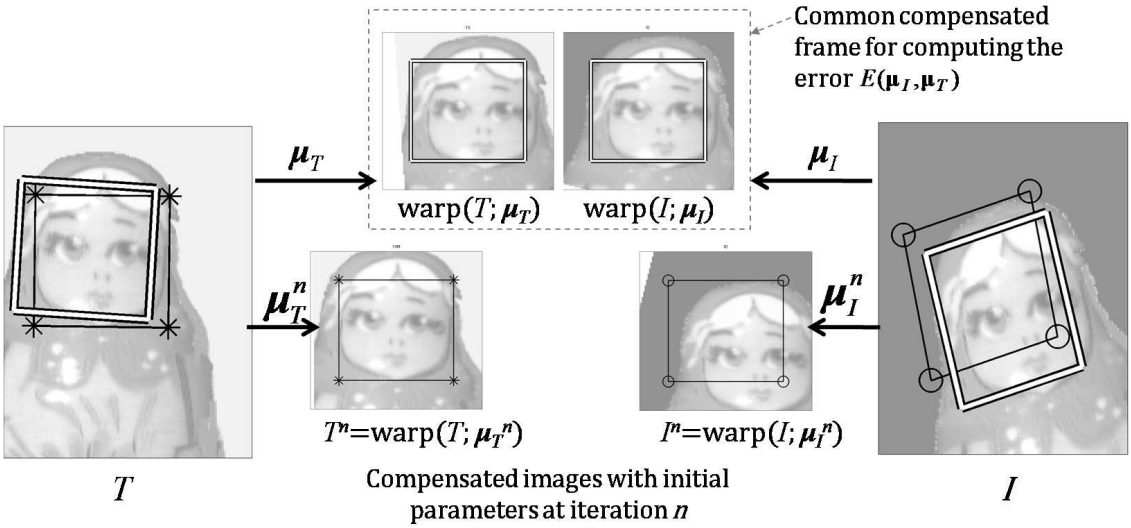

Fig. 2. General principle of the bidirectional framework, giving an overview of all the considerated coordinate frames. Region R is shown on $T(\mathbf{W}(\boldsymbol{\mu}_T^n,\mathbf{x}))$ and $I(\mathbf{W}(\boldsymbol{\mu}_I^n,\mathbf{x}))$ (corresponds to the initial $\mathbf{e}(\boldsymbol{\mu}_I^n,\boldsymbol{\mu}_T^n)$), as well as on $T(\mathbf{W}(\boldsymbol{\mu}_T,\mathbf{x}))$ and $I(\mathbf{W}(\boldsymbol{\mu}_I,\mathbf{x}))$ (corresponds to $\mathbf{e}(\boldsymbol{\mu}_I,\boldsymbol{\mu}_T)$ at convergence). These regions are warped onto I and T. For the Forwards approach, $\boldsymbol{\mu}_T$ and $\boldsymbol{\mu}_T^n$ are equal. For the Inverse approach, $\boldsymbol{\mu}_I$ and $\boldsymbol{\mu}_I^n$ are equal. In the general case shown here, the common coordinate frame corresponds to a compensation from both I and T using respectively $\boldsymbol{\mu}_I$ and $\boldsymbol{\mu}_T$.

Using the previous notations, we have:

$$J^{\mathrm{FCL}} = J_I^{CL}(0) \quad (27)$$
$$J^{\mathrm{ICL}} = J_T^{CL}(0) \quad (28)$$
$$J^{\mathrm{SCL}} = \frac{1}{2}(J_I^{CL}(0) + J_T^{CL}(0)) \quad (29)$$
$$J^{\mathrm{BCL}} = \left[J_I^{CL}(0) \mid J_T^{CL}(0)\right] \quad (30)$$

In practice, the Jacobian matrices are computed only for $\delta\mathbf{v} = 0$ using the chain rule:

$$J_{I,i}^{CL}(0) = \nabla I^n(\mathbf{x}_i)\left.\frac{\partial \mathbf{W}(\exp(\delta\mathbf{v}),\mathbf{x}_i)}{\partial \delta\mathbf{v}}\right|_0 \quad (31)$$
$$J_{T,i}^{CL}(0) = \nabla T^n(\mathbf{x}_i)\left.\frac{\partial \mathbf{W}(\exp(\delta\mathbf{v}),\mathbf{x}_i)}{\partial \delta\mathbf{v}}\right|_0 \quad (32)$$

where:

$$I^n(\mathbf{y}) = I(\mathbf{W}(\boldsymbol{\mu}_I^n,\mathbf{y})) \quad (33)$$
$$T^n(\mathbf{y}) = T(\mathbf{W}(\boldsymbol{\mu}_T^n,\mathbf{y})) \quad (34)$$

represent the warped image and template using initial estimates $\boldsymbol{\mu}_I^n$ and $\boldsymbol{\mu}_T^n$. This approach yields a computational gain, since these warped images have already been computed in order to obtain $\mathbf{e}$. Note that the group action property (5) is required to get benefit from this speedup.

D. Parameter update rule

When I and T are equal up to the compensation using the true parameters $(\bar{\boldsymbol{\mu}}_I,\bar{\boldsymbol{\mu}}_T)$ then an infinity of parameter pairs corresponding to their orbit $\mathrm{orbit}(\bar{\boldsymbol{\mu}}_I,\bar{\boldsymbol{\mu}}_T)$ w.r.t. right composition with the same parameter $\Delta\mu$ should also be considered as correct parameters:

$$\mathrm{orbit}(\boldsymbol{\mu}_I,\boldsymbol{\mu}_T) = \{(\boldsymbol{\mu}_I \circ \Delta\mu, \boldsymbol{\mu}_T \circ \Delta\mu) \quad | \quad \Delta\mu \in \mathcal{P}\}. \quad (35)$$

The update rule introduced in equation (17) should therefore select its result within the orbit of its input parameters.

A first solution is to use the identity update, which keeps the estimated parameters unmodified in the Bidirectional parameter space:

$$\mathrm{update}^{Id}(\boldsymbol{\mu}_I,\boldsymbol{\mu}_T) = (\boldsymbol{\mu}_I,\boldsymbol{\mu}_T). \quad (36)$$

This approach corresponds to a standard iterative optimization in the bidirectional parameter space, which is constrained to a specific subspace in the case of the FCL, ICL and SCL approaches. The convergences properties of the optimization algorithm applied at each iteration are then kept. In our case, this corresponds to the Gauss-Newton optimization of E on Lie Group.

When computational time is a concern, updating both parameters $\boldsymbol{\mu}_I$ and $\boldsymbol{\mu}_T$ requires to warp both images and compute their gradient at each new iteration, yielding a slower algorithm. To address this issue, a Forward update rule is used:

$$\mathrm{update}^{F}(\boldsymbol{\mu}_I,\boldsymbol{\mu}_T) = (\boldsymbol{\mu}_I \circ \boldsymbol{\mu}_T^{-1} \circ \boldsymbol{\mu}_T^0, \boldsymbol{\mu}_T^0). \quad (37)$$

The parameters $\boldsymbol{\mu}_T$ are then reinitialized to their original values $\boldsymbol{\mu}_T^0$ at each iteration. This approach presents two advantages.

First, J_T can be precomputed for all iterations, thus improving the speed of the algorithm [2].

Second, reinitializing $\boldsymbol{\mu}_T^n$ to $\boldsymbol{\mu}_T^0$ allows us to keep the region of interest R close to the true location of the object of interest in the template, and to avoid taking into account clutter in image I when the optimization comes close to the optimum. This is illustrated in Figure 3, where the valley formed around the manifold of correct parameters $\mathrm{orbit}(\bar{\boldsymbol{\mu}}_I,\bar{\boldsymbol{\mu}}_T)$ has a low error $E(\boldsymbol{\mu}_I,\boldsymbol{\mu}_T)$ in the bidirectional space $\mathcal{P}^2$ only for $\boldsymbol{\mu}_T$ close enough to $\bar{\boldsymbol{\mu}}_T$.

From the point of view of convergence, the value of the objective function may change during the update rule, as illustrated in Figure 3 by a jump along the orbit of the estimated parameters at the end of each iteration. In practice

the stopping criterion is therefore defined as the decrease between $E(\boldsymbol{\mu}_I^n, \boldsymbol{\mu}_T^n)$ and $E(\boldsymbol{\mu}_I^{n+1}, \boldsymbol{\mu}_T^{n+1})$, i.e. after the update.

This does not guarantee that the stopping criterion is not applied too early due to of a difference between the estimated $E(\hat{\boldsymbol{\mu}}_I, \hat{\boldsymbol{\mu}}_T)$ at step n and its updated value $E(\boldsymbol{\mu}_I^{n+1}, \boldsymbol{\mu}_T^{n+1})$. A natural choice to make those two values identical, up to the drift of the region of interest, would be to use an invariant objective function [14] [15], which takes into account volume forms compensation. This approach yields additional Jacobian terms in J which compensate for the bias associated to scale change when passing from one coordinate frame to another. This better theoretical invariance therefore comes at the cost of additional online computation, and will be left out of the scope of the present paper. We refer the reader to [15] for an in depth discussion of volume compensation issues.

For our application to parametric image alignment, the evoqued bias does not appear to be a major concern. Indeed, the application of such a Forwards update rule combined with the objective function (41) with the Forwards update rule was applied succcesfully in the Inverse Compositional algorithm [2] for which Baker and Matthews proposed an equivalence proof with Forwards compositional approaches. Benhimane and Malis [5] showed that the ESM algorithm converges as a second-order minimization of the Forwards objective function under assumption (10). As explained in next subsection this therefore also applies to the SCL algorithm with Forwards update rule.

E. Equivalence between the ESM *algorithm and the* SCL *approach*

We now discuss the relationship between the SCL approach that was introduced in section III and the ESM algorithm [4], which will provide a theoretical framework useful to better understand the advantages of the novel BCL algorithm which we will present in the next section.

The ESM algorithm is based on the second-order approximation of the vectorial error at the true parameters $\delta\mathbf{v}^*$, such that $\|\mathbf{e}(\delta\mathbf{v}^*)\|$ is null if (10) is satisfied:

$$\mathbf{e}(\delta\mathbf{v}^*) = \mathbf{e}(0) + \frac{1}{2}(J(0) + J(\delta\mathbf{v}^*))\delta\mathbf{v}^* + O(\|\delta\mathbf{v}^*\|^3). \quad (38)$$

One ESM iteration is obtained by setting $J = J^{FCL}(0) + J^{FCL}(\delta\mathbf{v}^*)$ in (24). This approximation is generic, but requires the Jacobian $J(\delta\mathbf{v}^*)$ at the true parameters to be known. Benhimane and Malis show in [5] that, when the assumption (10) holds, using a parameterization on Lie Groups yields:

$$J_I^{CL}(\delta\mathbf{v}^*)\delta\mathbf{v}^* = J_T^{CL}(0)\delta\mathbf{v}^*. \quad (39)$$

They therefore define an alternative Jacobian J_{ESM}, which is to be used by substituting $J^{\mathrm{FCL}}(\delta\mathbf{v}^*) = J_I^{CL}(\delta\mathbf{v}^*)$ with $J_T^{CL}(0)$ in the FCL version $\mathbf{e}^{\mathrm{FCL}}$ of (24):

$$J_{\mathrm{ESM}} = \frac{1}{2}(J_I^{CL}(0) + J_T^{CL}(0)). \quad (40)$$

J_{ESM} appears to be exactly the same matrix as $J^{\mathrm{SCL}}(0)$ defined in equation (29). Furthermore, the Forwards update rule $\boldsymbol{\mu}_I^{n+1} = \boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v})$ associated to the SCL approach is the same as the one used in the ESM algorithm.

This relationship provides new perspectives on the ESM algorithm. Indeed, when the assumption (10) does not hold, the equation (39) of the ESM algorithm is only approximative. The previous analysis reveals that the optimization of a FCL error using the ESM scheme (FCL-ESM) corresponds exactly to the optimization of a SCL error using a Gauss-Newton scheme (SCL-GN) with Forwards update rule.

IV. Bidirectional Compositional Lie algorithm

The Bidirectional approach stands apart from all other approaches, since it considers a bidirectionnal local reparameterization $(\delta\mathbf{v}_I, \delta\mathbf{v}_T)$, which hold the complete expression power of the bidirectional formulation that was introduced in this paper.

We will now focus on the BCL approach, based on the vectorial error $\mathbf{e}^{\mathrm{BCL}}(\delta\mathbf{v}_I, \delta\mathbf{v}_T)$ with

$$\begin{aligned} e_i^{\mathrm{BCL}}(\delta\mathbf{v}_I, \delta\mathbf{v}_T) = I(\mathbf{W}(\boldsymbol{\mu}_I \circ \exp(\delta\mathbf{v}_I), \mathbf{x}_i)) \\ -T(\mathbf{W}(\boldsymbol{\mu}_T \circ \exp(-\delta\mathbf{v}_T), \mathbf{x}_i)). \end{aligned} \quad (41)$$

With Gauss-Newton optimization, the estimation uses a Jacobian $J^{\mathrm{BCL}} = \begin{bmatrix} J_I & J_T \end{bmatrix}$ obtained by concatenating the contributions of both I and T:

$$\begin{bmatrix} \delta\mathbf{v}_I \\ \delta\mathbf{v}_T \end{bmatrix} = \begin{bmatrix} J_I & J_T \end{bmatrix}^+ \mathbf{e}^{\mathrm{BCL}}(0,0). \quad (42)$$

The associated update rule is the following:

$$\boldsymbol{\mu}_I^{n+1} = \boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v}_I) \circ \exp(\delta\mathbf{v}_T). \quad (43)$$

A. Invariance properties

A SCL approach corresponds to a BCL approach, where at each iteration the $2p$-dimensional local increment $(\delta\mathbf{v}_I, \delta\mathbf{v}_T)$ is constrained to explore the p-dimensional vector subspace corresponding to $\delta\mathbf{v}_I = \delta\mathbf{v}_T$ (as illustrated in Figure 3). A FCL approach explores the $(\delta\mathbf{v}_I, 0)$ subspace, an ICL approach the $(0, \delta\mathbf{v}_T)$ subspace. The BCL approach is not restricted to a specific subspace, and is therefore able to consider the solutions provided by the FCL, the ICL or the SCL subspaces. Since the ESM/SCL algorithm was shown [5] to have the best theoretical convergence properties, we will now discuss the interpretation of the additional dimensions that are orthogonal to the SCL subspace, and that the BCL is able to explore.

Let us consider the following change of variable:

$$\delta\mathbf{v}_\oplus = \delta\mathbf{v}_I + \delta\mathbf{v}_T \quad \text{and} \quad \delta\mathbf{v}_\ominus = \delta\mathbf{v}_I - \delta\mathbf{v}_T \quad (44)$$

where $\delta\mathbf{v}_\oplus$ and $\delta\mathbf{v}_\ominus$ stand for the symmetric and anti-symmetric parts respectively.

The previous change in variable (44) can be interpreted by noting that to the first order to $\delta\mathbf{v}_I$ and $\delta\mathbf{v}_T$:

$$\begin{aligned} \mathbf{e}^{\mathrm{BCL}}(\delta\mathbf{v}_I, \delta\mathbf{v}_T) &\approx \mathbf{e}^{\mathrm{BCL}}(0,0) + J_I^{CL}(0)\delta\mathbf{v}_I + J_T^{CL}(0)\delta\mathbf{v}_T \\ &\approx \frac{1}{2}\mathbf{e}^{\mathrm{BCL}}(0,0) + \frac{1}{2}J_I^{CL}(0)\delta\mathbf{v}_\oplus + \frac{1}{2}J_T^{CL}(0)\delta\mathbf{v}_\oplus \\ &\quad + \frac{1}{2}\mathbf{e}^{\mathrm{BCL}}(0,0) + \frac{1}{2}J_I^{CL}(0)\delta\mathbf{v}_\ominus - \frac{1}{2}J_T^{CL}(0)\delta\mathbf{v}_\ominus \\ &\approx \frac{1}{2}(\mathbf{e}^{\mathrm{BCL}}(\delta\mathbf{v}_\oplus, \delta\mathbf{v}_\oplus) + \mathbf{e}^{\mathrm{BCL}}(\delta\mathbf{v}_\ominus, -\delta\mathbf{v}_\ominus)). \end{aligned} \quad (45)$$

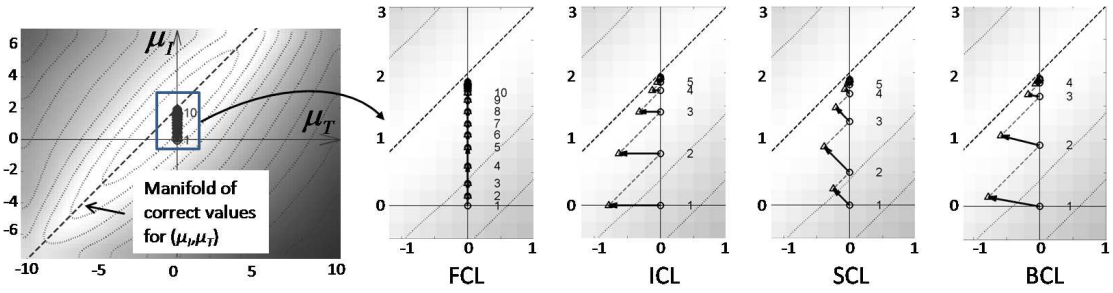

Fig. 3. Illustration of the error function $E(\boldsymbol{\mu}_I, \boldsymbol{\mu}_T)$ and the trajectories of the iterations of the FCL, ICL, SCL and BCL approaches during the alignment of the two images of figure 2, where the true deformation is a 2 pixels horizontal translation $\bar{\boldsymbol{\mu}}_1 = 2$. Only the horizontal translation coefficients $(\boldsymbol{\mu}_{I,1}, \boldsymbol{\mu}_{T,1})$ are shown, which corresponds to a slice of the bidirectional space $\mathcal{P}^2$. For the error E, darker shade of gray means higher error value. Translation was estimated using GN optimization with Forwards update after each iteration. Each iteration n is plotted as an arrow that links $(\boldsymbol{\mu}_I^n, \boldsymbol{\mu}_T^n)$ (numbered $\odot$ bullets) to $(\boldsymbol{\mu}_I(\boldsymbol{\mu}_I^n, \delta\hat{\mathbf{v}}), \boldsymbol{\mu}_T(\boldsymbol{\mu}_T^n, \delta\hat{\mathbf{v}}))$ ($\triangle$ bullets), and a dashed segment that links to the next iteration $(\boldsymbol{\mu}_I^{n+1}, \boldsymbol{\mu}_T^{n+1})$ using the Forwards update rule. The initialisation is $(\boldsymbol{\mu}_{I,1}^0, \boldsymbol{\mu}_{T,1}^0) = (0, 0)$. The dashed line $(\boldsymbol{\mu}_{I,1}, \boldsymbol{\mu}_{T,1}) = (\boldsymbol{\mu}_1, 2 + \boldsymbol{\mu}_1)$ represents the manifold of correct estimates.

The first term corresponds to the standard SCL error, where $\delta\mathbf{v}_I = \delta\mathbf{v}_T = \frac{1}{2}\delta\mathbf{v}_\oplus$, i.e. the images are compensated symmetrically towards each other. The second term is specific to the BCL approach. It corresponds to the error when compensating the two images within the orbit of the initial parameters $(\boldsymbol{\mu}_I^n, \boldsymbol{\mu}_T^n)$ defined in section III-D, where the relative transformation $\boldsymbol{\mu}_I \circ \boldsymbol{\mu}_T^{-1}$ remains constant.

The linearization of the error with respect to $(\delta\mathbf{v}_\oplus, \delta\mathbf{v}_\ominus)$ is obtained from (45):

$$\mathbf{e}^{\mathrm{BCL}}(\delta\mathbf{v}_I, \delta\mathbf{v}_T) \approx \mathbf{e}^{\mathrm{BCL}}(0,0) + J_\oplus\delta\mathbf{v}_\oplus + J_\ominus\delta\mathbf{v}_\ominus, \quad (46)$$

with

$$\begin{aligned} J_\oplus &= \frac{1}{2}(J_I^{CL}(0) + J_T^{CL}(0)) \text{ and} \\ J_\ominus &= \frac{1}{2}(J_I^{CL}(0) - J_T^{CL}(0)). \end{aligned} \quad (47)$$

The minimization of $E(\delta\mathbf{v}_I, \delta\mathbf{v}_T)$ in terms of the new variables $(\delta\mathbf{v}_\oplus, \delta\mathbf{v}_\ominus)$ therefore corresponds to:

$$\begin{bmatrix}\delta\mathbf{v}_\oplus \\ \delta\mathbf{v}_\ominus\end{bmatrix} = \begin{bmatrix}J_\oplus & J_\ominus\end{bmatrix}^+ \mathbf{e}^{\mathrm{BCL}}(0,0), \quad (48)$$

At this stage, since $\delta\mathbf{v}_\oplus$ and $\delta\mathbf{v}_\ominus$ can be assumed to be small increment, using the Campbell-Baker-Hausdorff formula [12] on (43) yields to:

$$\boldsymbol{\mu}_I^{n+1} \approx \boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v}_\oplus). \quad (49)$$

It follows that $\delta\mathbf{v}_\ominus$ is a parameter which is modeled in the estimation but which value does not modify the final result. This principle is similar to the use of parameters modeling illumination change [16] [17], in order to make the estimation of motion parameters invariant to such perturbations. The invariance that the BCL algorithm enforces at each iteration is instead a local invariance around $\delta\mathbf{v} = 0$ with respect to the $J_\ominus$ subspace, that corresponds to a change of the reference coordinate frame. This property is at the core of the PBCL algorithm.

B. The Projected BCL *algorithm (*PBCL*)*

The Projected BCL algorithm (PBCL) uses the approximation rule (49) in order to avoid solving equation (24) with the full bidirectional Jacobian matrix J^{BCL}.

Due to the properties of block matrix pseudo-inverse when $\begin{bmatrix}J_\oplus & J_\ominus\end{bmatrix}$ is full-rank:

$$\delta\mathbf{v}_\oplus = \left(P_\ominus^\perp J_\oplus\right)^+ \mathbf{e}^{\mathrm{BCL}}(0,0), \quad (50)$$

where $P_\ominus^\perp$ represents the projection within the N-dimensional error vector space onto the orthogonal subspace to the column span of $J_\ominus$:

$$P_\ominus^\perp = Id_N - J_\ominus({J_\ominus}^t J_\ominus)^{-1}{J_\ominus}^t. \quad (51)$$

The difference between the PBCL and the SCL approaches consists in using $P_\ominus^\perp J_\oplus$ instead of $J_\oplus$, i.e. projecting out the dimensions in the error space that correspond to a bidirectional compensation that changes the reference frame without changing the relative transformation between I and T.

Furthermore, by noting that we have on the one hand $J^{FCL} = J_\oplus + J_\ominus$, $J^{ICL} = J_\oplus - J_\ominus$ and $J^{SCL} = J_\oplus$, and on the other hand $P_\ominus^\perp J_\ominus = 0$, we get:

$$P_\ominus^\perp J^{\mathrm{FCL}} = P_\ominus^\perp J^{\mathrm{ICL}} = P_\ominus^\perp J^{\mathrm{SCL}} = P_\ominus^\perp J_\oplus. \quad (52)$$

Indeed, the differences between the FCL, ICL and SCL approaches depend on a differently weighted contribution of $J_\ominus$, which is removed by the projection $P_\ominus^\perp$. This property allows us to define without any lack of genericity the Jacobian associated to the PBCL approach with respect to a precomputed J^{ICL}, without computing $J_\oplus$ explicitely:

$$J^{\mathrm{PBCL}} = P_\ominus^\perp J^{\mathrm{ICL}}. \quad (53)$$

In the noise free case, assuming (10) holds and a Lie Group parameterization is used, projecting the fundamental equation of the ESM algorithm (38) yields, thanks to (52),

$$P_\ominus^\perp \mathbf{e}(\delta\mathbf{v}^*) = P_\ominus^\perp \mathbf{e}(0) + J^{\mathrm{PBCL}}\delta\mathbf{v}^* + O(\|\delta\mathbf{v}^*\|^3). \quad (54)$$

The difference with the original ESM is that it is expressed in the orthogonal subspace of the column-span of $J_\ominus$, which enforces the local invariance of the objective fonction with respect to a change in the reference coordinates frame.

C. Theoretical analysis of the effect of noise

Let us consider two noisy images with independant noises:

$$I = I_{nf} + \epsilon_I \qquad T = I_{nf} + \epsilon_T \tag{55}$$

which impacts both the zeroth-order and the first-order terms:

$$\begin{aligned} \mathbf{e}(0) &= \mathbf{e}_{nf}(0) + \epsilon_I - \epsilon_T & (56)\\ J_{\oplus} &= J_{\oplus nf} + \frac{1}{2}(J_{\epsilon_I} + J_{\epsilon_T}) & (57)\\ J_{\ominus} &= J_{\ominus nf} + \frac{1}{2}(J_{\epsilon_I} - J_{\epsilon_T}), & (58) \end{aligned}$$

where:

$$\begin{aligned} J_{\epsilon_I i} &= \frac{\partial \epsilon_I(\mathbf{W}(\boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v})), \mathbf{x}_i)}{\partial \delta\mathbf{v}} & (59)\\ J_{\epsilon_T i} &= \frac{\partial \epsilon_T(\mathbf{W}(\boldsymbol{\mu}_I^n \circ \exp(\delta\mathbf{v})), \mathbf{x}_i)}{\partial \delta\mathbf{v}}. & (60) \end{aligned}$$

In the noisy case, the SCL error (38) becomes a first-order approximation, with a noisy Jacobian:

$$\mathbf{e}(\delta\mathbf{v}^*) = \mathbf{e}(0) + (J_{\oplus nf} + \frac{1}{2}(J_{\epsilon_I} + J_{\epsilon_T}))\delta\mathbf{v}^* + O(\|\delta\mathbf{v}^*\|^2). \tag{61}$$

If we denote by η_I^2 and η_T^2 the variances of the noise in $J_{\epsilon_I}\delta\mathbf{v}^*$ and $J_{\epsilon_T}\delta\mathbf{v}^*$. The variance of the noise in the ESM Jacobian $J_{\oplus}\delta\mathbf{v}^*$ is therefore $\frac{1}{4}(\eta_I^2+\eta_T^2)$, which is half the noise variance of $J^{\mathrm{FCL}}\delta\mathbf{v}^*$ or $J^{ICL}\delta\mathbf{v}^*$ for symmetric levels of noise.

Projecting (61), the projected PBCL error (54) also becomes a first-order approximation in the noisy case:

$$P_{\ominus}^{\perp}\mathbf{e}(\delta\mathbf{v}^*) = P_{\ominus}^{\perp}\mathbf{e}(0) + P_{\ominus}^{\perp}J_{\oplus}\delta\mathbf{v}^* + O(\|\delta\mathbf{v}^*\|^2). \tag{62}$$

We now show that $P_{\ominus}^{\perp}$ projects out a part of the noise from the Jacobian $J_{\oplus}$. Using equation (39), we get:

$$\begin{aligned} J_{\ominus nf}\delta\mathbf{v}^* &= (J_{Inf}(0) - J_{Tnf}(0))\delta\mathbf{v}^* & (63)\\ &= (J_{Inf}(0) - J_{Inf}(\delta\mathbf{v}^*))\delta\mathbf{v}^* = \mathcal{O}(\|\delta\mathbf{v}^*\|^2). & (64) \end{aligned}$$

Moreover, by definition $P_{\ominus}^{\perp}J_{\ominus} = 0$, which yields:

$$\begin{aligned} P_{\ominus}^{\perp}(J_{\epsilon_I} - J_{\epsilon_T})\delta\mathbf{v}^* &= 2P_{\ominus}^{\perp}(J_{\ominus} - J_{\ominus nf})\delta\mathbf{v}^* & (65)\\ &= 2P_{\ominus}^{\perp}J_{\ominus nf}\delta\mathbf{v}^* = \mathcal{O}(\|\delta\mathbf{v}^*\|^2). & (66) \end{aligned}$$

Let us assume the template is less corrupted than the image. Using (66), we can modify the expression of $P_{\ominus}^{\perp}J_{\oplus}$ as follows:

$$\begin{aligned} P_{\ominus}^{\perp}J_{\oplus}\delta\mathbf{v}^* &= P_{\ominus}^{\perp}(J_{\oplus nf} + J_{\epsilon_I} - J_{\epsilon_T} + 2J_{\epsilon_T})\delta\mathbf{v}^* & (67)\\ &= P_{\ominus}^{\perp}(J_{\oplus nf} + 2J_{\epsilon_T})\delta\mathbf{v}^* + \mathcal{O}(\|\delta\mathbf{v}^*\|^2). & (68) \end{aligned}$$

In case there is asymmetry in the noise levels, equation (68) hints that the influence of the noise on the Jacobian matrix $J_{\oplus}$ can be reduced within the $P_{\ominus}^{\perp}$ subspace. Indeed, from the full error $\frac{1}{2}(J_{\epsilon_I} + J_{\epsilon_T})$, only the less noisy part J_{ϵ_T} is kept. In particular, no corruption of the projected Jacobian should be observed when the template is noiseless.

The theoretical results derived in this subsection will be validated experimentally in paragraphs V-B2 and V-B3.

V. Experimental comparison of the approaches

In the following the performances of the new algorithms PBCL and BCL are compared to the conventional FCL, ICL and ESM approaches. This study takes into account the computational complexity, the rate of convergence and the frequency of convergence. Both simulated warps and real video data have been used.

All studied methods are based on the Gauss Newton scheme (24) with Forwards update rule (37). The ESM fits into this framework as an SCL approach according to subsection III-E.

A. Computational complexity

The generic algorithmic scheme is the following:

- Pre-computation
  1 Warp T to compute T^0.
  2 Compute the gradients of T^0 : ∇T^0.
  3 Evaluate the complete jacobian of T : $J_T^{CL}(0)$.
  4 Only for the ICL, precompute the pseudo-inverse of the jacobian.
- For each iteration
  5 Warp I with $\mathbf{W}(\boldsymbol{\mu}_I^n, \mathbf{x})$ to compute I^n.
  6 Compute the error $\mathbf{e}(0)$.
  7 Compute the gradient ∇I^n of the warped image.
  8 Evaluate the complete jacobian of I : $J_I^{CL}(0)$.
  9 Compute $J(0)$.
  10 Solve $J(0)\delta\mathbf{v} = \mathbf{e}(0)$.
  11 Update the warp parameters with (37).

where:
$J = J_I^{CL}(0)$ for the FCL algorithm,
$J = J_T^{CL}(0)$ for the ICL algorithm,
$J = \frac{1}{2}(J_I^{CL}(0) + J_T^{CL}(0))$ for the ESM algorithm,
$J = [J_I^{CL}(0) \mid J_T^{CL}(0)]$ for the BCL algorithm,
$J = P_{\ominus}^{\perp}J_{\oplus}$ for the PBCL algorithm.
Note that for FCL, steps 2, 3 and 4 are skipped. For ICL, steps 7, 8 and 9 are skipped.

In the following, we note p the number of parameters $\{\delta\mathbf{v}_k\}_{k=\{1..p\}}$ and N the number of image pixels. The asymptotic complexity of the different algorithms can be found in table I for $p << N$. Step 10 was coded using the pseudo-inverse calculation proposed in [2]. The computational cost can be found in table II for our current Matlab implementation on a Intel(R) Core(TM)2 duo CPU 3.0 GHz with 4 GB of RAM. A 100x100 patch is used to estimate a homography parameterized as in [5].

As shown in [2], ICL is the more computationally efficient algorithm because it does not need the image Jacobian $J_I^{CL}(0)$ to be computed (steps 6, 7, 8). Thus it is possible to pre-compute the pseudo-inverse $A = (J_T^{CL}(0))^+$ and obtain $\delta\mathbf{v}$ by a matrix product $\delta\mathbf{v} = Ae(0)$ (step 9).

The FCL and ESM algorithms are in practice equivalent in terms of computational cost (when many iterations are done) because the extra heavy computations of the ESM can be done as pre-computations.

The PBCL algorithm has globally a similar asymptotic complexity, but requires an additional projection $(P_{\ominus}^{\perp}J_{\oplus})$ to be computed. In practice, it can be performed more efficiently using a QR decomposition of $J_{\ominus}$ but it remains quite costly.

Algo	Pre-computation Steps				
	1	2	3	4	Total
FCL	pN	–	–	–	pN
ICL	pN	N	pN	p^2N	p^2N
Others	pN	N	pN	–	pN

Algo	Iterative Steps							
	5	6	7	8	9	10	11	Total
FCL	pN	N	N	pN	–	p^2N	p^2	p^2N
ICL	pN	N	–	–	–	pN	p^2	pN
ESM	pN	N	N	pN	pN	p^2N	p^2	p^2N
BCL	pN	N	N	pN	–	p^2N	p^2	p^2N
PBCL	pN	N	N	pN	p^2N	p^2N	p^2	p^2N

TABLE I
ASYMPTOTIC COMPUTATIONAL COMPLEXITY OF CONVENTIONAL AND NEW IMAGE ALIGNMENT ALGORITHMS FOR p PARAMETERS AND N PIXELS (WITH $p << N$).

Algo	Pre-computation Steps				
	1	2	3	4	Total
FCL	4.25	–	–	–	4.25
ICL	4.21	0.44	1.70	1.22	7.57
Others	4.23	0.43	1.67	0	6.33

Algo	Iterative Steps							
	5	6	7	8	9	10	11	Total
FCL	4.42	0.02	0.43	1.70	0	0.40	0.15	7.12
ICL	4.40	0.02	0	0	0	0.08	0.15	4.65
ESM	4.42	0.01	0.43	1.70	0.71	0.40	0.15	7.82
BCL	4.44	0.01	0.43	1.80	0	1.17	0.28	8.13
PBCL	4.43	0.01	0.44	1.65	3.08	0.40	0.15	10.25

TABLE II
COMPUTATIONAL TIME (IN MILLISECONDS) OF CONVENTIONAL AND NEW IMAGE ALIGNMENT ALGORITHMS WITH $p = 8$, $N = 10000$. FOR ITERATIVE STEPS, THE TIME IS GIVEN FOR ONE ITERATION.

The BCL algorithm has the same asymptotic complexity as the ESM algorithm, but requires solving a system with twice the number of parameters, which makes step 9 more costly. This is compensated by the fact that the jacobian matrices do not need to be added in step 8. The total time is therefore only slightly larger.

B. Convergence and robustness evaluation

The benchmark used is inspired by the one proposed in [2]. Random disturbances are generated by adding a spatial Gaussian noise of standard deviation σ_{noise} (called Point Sigma) to four canonical point locations in a reference image I_{ref}: these four pairs of points (canonical and test points) define an homography warp parameter vector for the disturbance. Using these parameters, the reference image I_{ref} is warped onto template image T_{ref}.

I_{ref} and T_{ref} are corrupted with additive gaussian noise: $I = I_{ref} + \epsilon_I$ and $T = T_{ref} + \epsilon_T$. The noise levels are characterized by their variances σ_I^2 and σ_T^2 and the corresponding Signal to Noise Ratios SNR_I and SNR_T with respect to the noise-free image:

$$SNR = 10 \log_{10} \left(\frac{\mathbb{E}({I_{ref}}^2)}{\sigma^2} \right). \quad (69)$$

The image alignment algorithms are then run in order to fit image I to image T. The convergence criteria is the root mean squared error of the distance between the test point locations and the destination locations of the canonical points warped with the estimated deformation (denoted RMS point error). Two main performance criteria are considered:

- Average frequency of convergence: percentage of tests where an algorithm converged to the correct estimate (defined as a RMS Point Error less than 1 pixel),
- Average rates of convergence: for tests that converge for all methods, the average RMS point error is plotted against the algorithm iteration number.

In the following, we present average results obtained with the five images shown in figure 4. The motion model is a homography parameterized as in [5]. For each algorithm and each test, 30 iterations are done. For average frequency of convergence, 500 tests are done per image and per Point Sigma. For Average rates of Conververgence, 100 tests are done per image.

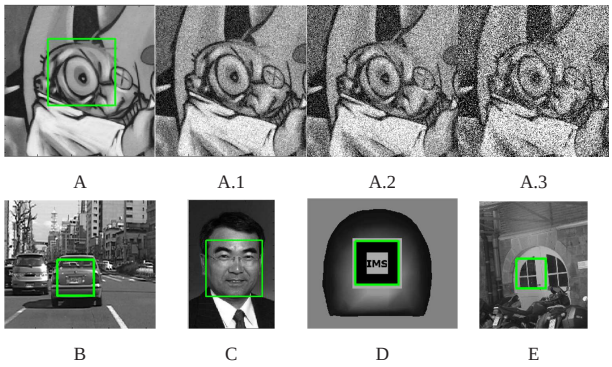

Fig. 4. Images used for the experiments. First row: The noise-free image (left) is used for extracting the template image. Image I is then obtained by adding gaussian noise to the noise free image. The noisy images (right) are shown with respective SNR: 15 dB, 10 dB and 5 dB. Second row: other images used for the experimentations. Images A and E come from the INRIA Learning and Recognition in Vision (LEAR) dataset (http://lear.inrialpes.fr/data). Images B and C come from the benchmark of Baker and Matthwews [2]. Image D was synthetized using an openGL implementation.

1) Frequency and rate of convergence: Figure 5 shows a comparison of the different approaches in three typical situations: no noise (left), noise on the image only, the template being noiseless (middle), and noise on both image (right).

The BCL and the PBCL exhibit identical performances for all tests. This confirms experimentally equation (49), and justifies that they will be assimilated in the discussion.

The performances in terms of frequency of convergence and rate of convergence (Fig. 5) present similar ranking. The BCL provide the best overall performance when noise is present, being only slightly less performant than the ESM in the noiseless case (left column).

In the case where the template is noiseless and the image corrupted by a SNR of 10dB (middle column), ICL is better than ESM at convergence rate and final precision, which illustrate the detrimental effect of noise in the Jacobian. The BCL approach converges faster and reaches the same accuracy as ICL, although BCL includes corrupted image gradients in the estimation. This illustrates the ability of our approach to project out the noise from the Jacobian when there is asymmetry in the noise levels, as expected from (68).

When noise is present on both image and template at

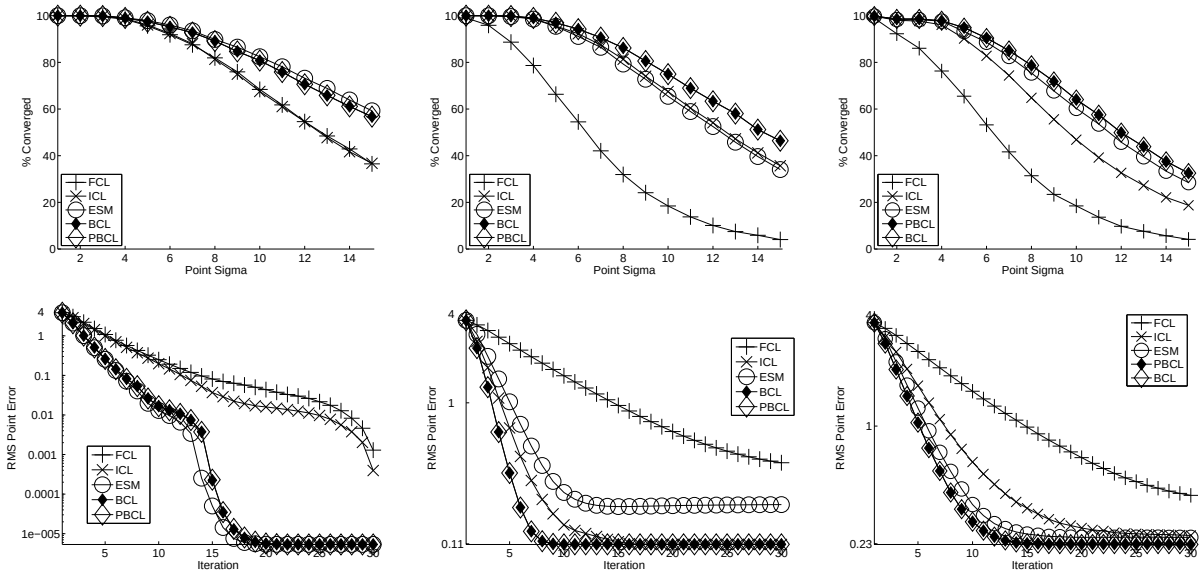

Fig. 5. Frequency of convergence (top row) and rate of convergence in logarithmic scale for a Point Sigma equal to 4 pixels (bottom row) for the FCL, ICL, SCL, BCL and PBCL approaches using a Gauss-Newton optimization with Forwards update rule. Left: I and T are noiseless. Middle: additive gaussian noise of SNR=10dB was added to I, T is noiseless. Right: additive gaussian noise was added on both images, SNR=10dB for I, SNR=15dB for T.

Algo	$P^{\perp}$	J	$\mathbf{e}$
ICL	Id_N	J_T^{CL}	$\mathbf{e}(0)$
ESM	Id_N	$J_{\oplus}$	$\mathbf{e}(0)$
PBCL	$P^{\perp}_{J_{\ominus}}$	$J_{\oplus}$	$\mathbf{e}(0)$
BCL	Id_N	$[J_I^{CL}\|J_T^{CL}]$	$\mathbf{e}(0)$
ESM_{nf}	Id_N	$J_{\oplus nf}$	$\mathbf{e}_{nf}(0)$
PBCL_{-}	$P^{\perp}_{J_{\ominus nf}}$	$J_{\oplus}$	$\mathbf{e}(0)$
PBCL_{ϵ}	$P^{\perp}_{J_{\epsilon}}$	$J_{\oplus}$	$\mathbf{e}(0)$

TABLE III
PROJECTION MATRIX $P^{\perp}$, JACOBIAN MATRIX J AND INITIAL ERROR $\mathbf{e}$ ASSOCIATED TO THE ALGORITHMS DEFINED IN EQUATION (70).

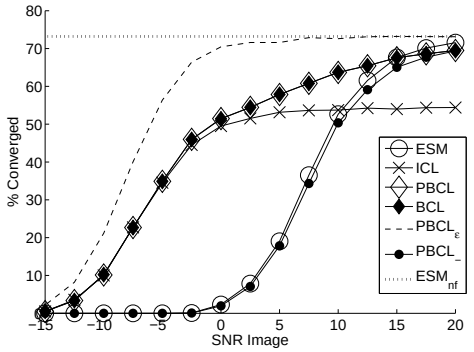

Fig. 6. Frequency of convergence for the ESM, ICL and PBCL algorithms compared to the synthetic PBCL_{-}, PBCL_{ϵ} and the ESM_{nf} algorithms for a Point Sigma equal to 12 and a decreasing image noise (-15dB to 20dB). No noise has been used on the template.

different levels (right column), BCL is slightly better than ESM. They both outperform the unidirectional approaches in this context.

2) Impact of the projection: In order to evaluate more accurately the impact of the projection $P^{\perp}_{\ominus}$ in the performance gain, we define three additional reference synthetic algorithms PBCL_{-}, PBCL_{ϵ} and ESM_{nf} which use noise-free data in some of their computations. All algorithms estimate $\delta\mathbf{v}$ using the following equation:

$$\delta\mathbf{v} = (P^{\perp}J)^{+}\ \mathbf{e}. \tag{70}$$

A summary of the specific values of the matrix $P^{\perp}$, J and $\mathbf{e}$ is shown in table III for the synthetic algorithms, as well as the standard ones. The figure 6 shows their performances with respect to image noise.

The synthetic PBCL_{ϵ} algorithm assumes the Jacobian J_{ϵ} of the noise ϵ is known, as defined in (59).

It provides the best performance amongst the methods that use a noisy error by projecting out the noise that corrupts $J_{\oplus}$. It has a similar performance as the ESM_{nf} over a large set of noise conditions, which confirms that the estimation is very sensitive to the presence of noise in the Jacobian matrix, and that the projection approach is relevant.

The synthetic PBCL_{-} algorithm uses the noise-free $P^{\perp}_{\ominus nf}$ projection. It provides approximatively the same performance as the ESM method, which shows that projecting by $P^{\perp}_{\ominus}$ has little undesirable effect on the performance. It is slighty less robust than ESM in low-noise conditions. The Jacobian $J_{\ominus}$ has therefore a small correlation with the spatial gradients of the images, which leads to projecting out a little amount of relevant information.

The PBCL algorithm provides the best performances within non synthetic algorithms over almost the full range of noise levels. Like the PBCL_{-}, it becomes slighty less robust than ESM in low-noise conditions. These results can be related to equation (68) to confirm that the PBCL approach uses a relevant projection, which has a slight side-effect in the noise-free case, but improves the robustness as soon as there is

asymmetry in the noise levels.

3) Impact of the noise asymmetry: The following experiments show the evolution of the performances with respect to noise asymmetry. Noise is parameterized by a variance σ^2, and an asymmetry coefficient $\beta \in [0,1]$. The amount of noise allocated to I and T is defined as follows:

$$\sigma_I^2 = (1-\beta)\sigma^2, \quad \sigma_T^2 = \beta\sigma^2, \quad \text{with } \beta \in [0,1]. \qquad (71)$$

where σ_I^2 (resp. σ_T^2) is the variance of the noise corrupting image I (resp. the template T). Additive white gaussian noise is used. Since the noises of the two images are independent, $\sigma^2 = \sigma_I^2 + \sigma_T^2$ represents the total noise variance that was injected into the difference image $(I-T)$. The corresponding total SNR is computed as in (69) with respect to the variance of the image.

The figure 7 shows the average frequency of convergence, with respect to β. The results are obtained on the images of fig. 4 for two levels of noise (SNR of 10dB and 5dB) and for two levels of imprecision in the initialization (Point sigma of 6 and 12 pixels).

We can observe that the FCL approach (resp. ICL) has a decreasing performance when β decreases (resp. increases) which corresponds to an increasing level of noise in the gradients of I (resp. T). For a fixed SNR, the ESM approach provides approximately similar results for all values of β. Indeed, the noise variance in $J^{\text{ESM}} = \frac{1}{2}(J_I + J_T)$ is $\frac{1}{4}$ of the sum of the noise variances in J_I and J_T, which is constant in this experiment. This lower variance combines with equation (38) to explain the much better performance of the ESM for $\beta = 0.5$ in the case of noise.

The ESM approach outperforms the FCL and ICL approaches for all tested conditions, excepted for high asymmetry of the noise levels. For a 10dB SNR, they have similar performances for noise-free template or image (Fig. 7 left and middle). This ranking is reversed only for both a high level of noise, i.e. SNR=5dB, and highly asymmetric noise levels (Fig. 7 right).

The BCL approach performs as good or better than the ESM, FCL and ICL. It has almost the same performance as the ESM for $\beta = 0.5$, and provides an increasing gain when the amount of noise asymmetry increases. This gain is small for low noise situations, where the ESM has already near perfect convergence performances, but is significative in more difficult conditions, involving a higher noise level or a farther initialization.

C. Application to tracking in a low-light environment

In low-light conditions, an optical imaging system produces bad quality images that can be modeled according to [18]: the observed number of photons at one pixel is drawn from a Poisson distribution whose parameter is proportional to the average received intensity. Thus tracking an object using gradient-based approach becomes a challenging task because the Poisson noise can severely corrupt the gradients of the images. In order to improve the tracking performance on this kind of data, one would try to lower the noise on the template by averaging several registered frames. This approach yields an asymmetric image alignment problem where the current image I is registered to the template T of higher quality. We now evaluate the usefulness of the proposed algorithms in such a context.

Fig. 8. Images used for the low-light condition experiments in paragraph V-C. First row: images extracted from the high quality video sequences from the "Fast Far" set of [19] available at http://www.metaio.com/research. Second row: Simulated low-light acquisitions.

Low-light video sequences with controled ground-truth and parameters are not publically available, or are subject to confidentiality use. We simulated low-light conditions by corrupting publicly available videos (see Fig. 8) taken in day light conditions [19]. The corruption procedure consists in assigning to $I(\mathbf{x})$ a random value drawn from a Poisson distribution of mean $(a \cdot I(\mathbf{x}) + b)$. The scaling factors a and b are chosen to rescale the image values in the range $[1, 10]$.

The groundtruth is constructed from the initial high-quality sequence. The region of interest is defined on the first frame of the video, and its content is initially used as template image T. The ESM algorithm is used to track the region of interest, using the same homography model as in subsection V-B. This estimation is used as ground truth (true parameters $\bar{\boldsymbol{\mu}}(t)$). We manually checked than the ESM managed to track the object of interest on those sequences.

The tests are performed on the corrupted sequences as follows. The performance analysis involves the same algorithms as in subsection V, run with 40 iterations. The template is computed on the first nine frames of the video sequence by averaging the compensated images. Motion estimation is performed from frame $t = 11$ to 100 using the averaged template and the current corrupted image: for each frame of the sequence the image alignment algorithm is reinitialized with $\boldsymbol{\mu}_I^n(t+1) = \bar{\boldsymbol{\mu}}(t)$ and $\boldsymbol{\mu}_T^0(t+1) = 0$. We compute the RMS Point error by comparing the four corners of the region of interest predicted by the tested algorithm and the ground-truth. In order to obtain significant results, we averaged the results on 7 video sequences from the "Fast Far" set, and generated 10 different corrupted sequences for each initial high-quality sequence, yielding a total of 6300 estimation results for each approach.

Figure 9 shows the cumulative distribution of the RMS Point Error obtained by the various approaches, which characterizes the frequency of convergence for various values of RMS Point error threshold τ. The statistics of these distributions for $\tau = 3\ pixels$ are summarized in table IV.

The BCL approach provides the best performances with both the lowest mean RMS Point Error, lowest standard deviation, and best frequency of convergence for any threshold τ. In terms of frequency of convergence at $\tau = 3\ pixels$, it is followed by the ESM algorithm and the ICL approach.

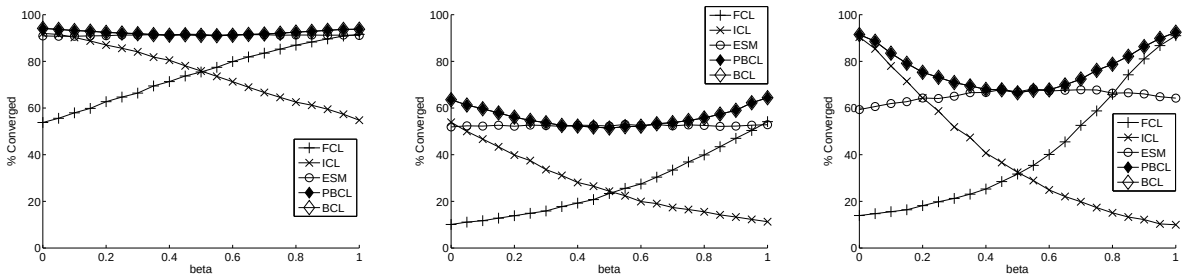

Fig. 7. Frequency of convergence for the FCL, ICL, ESM, PBCL and BCL algorithms with respect to noise asymmetry. Left: SNR=10dB, point sigma = 6 pixels. Middle: SNR=10dB, point sigma = 12 pixels. Right: SNR=5dB, point sigma = 6 pixels

$Algo$	FCL	ICL	ESM	BCL	$Init$
$freq.\ conv.$	93.1%	93.4%	98.5%	**99.1%**	
$mean$	1.2813	0.5711	0.6784	**0.4216**	3.2620
std	0.6300	0.3150	0.3782	**0.2835**	1.7023
$\frac{std}{\sqrt{nb_{test}}}$	0.0082	0.0040	0.0048	0.0036	

TABLE IV
FREQUENCY OF CONVERGENCE AND ACCURACY EXPRESSED AS MEAN AND STANDARD DEVIATION OF THE RMS POINT ERROR OBTAINED FOR THE EXPERIMENT OF PARAGRAPH V-C. ONLY TESTS THAT CONVERGED FOR $\tau = 3\ pixels$ ARE TAKEN INTO ACCOUNT. THE INITIAL RMS POINT ERROR STATISTICS IS ALSO PROVIDED AS REFERENCE.

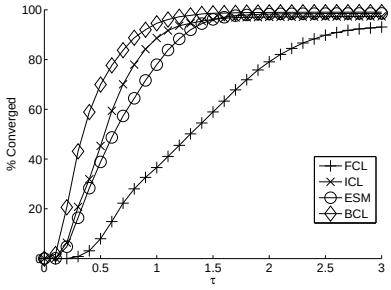

Fig. 9. Frequency of convergence for the experiment of paragraph V-C with respect to the threshold τ on the RMS Point Error.

In terms of average accuracy the ICL algorithm is better than the ESM algorithm. By computing its gradients only on the noisier image I, the FCL approach provides the worst results. These promising results concerning the BCL illustrate the perspectives of application of the proposed approach to improve image alignment results on real world low-SNR image data.

D. Experimental conclusion and recommendations

From the point of view of computational complexity (see tab. II and I), the ICL approach is the most efficient. But this criterion has to be balanced with respect to the convergence properties, which are rather in favor of approaches that take information from both of the image and the template. The ESM is only slightly slower than the FCL with much improved converge properties. The BCL approach is itself a little bit slower that the ESM.

An ICL approach should be prefered when one of the images is almost noise-free and the other image is severely corrupted by noise. The ESM or the BCL approach should be prefered in most other situations. When the level of noise is low, or when both images have the same level of noise, the ESM would be slightly faster, for a slightly better robustness. The BCL approach should be more robust on a range of noise conditions, and should provide better performances for intermediate asymmetry of noise, particularly in difficult conditions.

VI. CONCLUSION

In this paper we have presented a novel formulation of image alignment based on the combination of the bidirectional compensation of both image and template and a Lie group parameterization of the motion. Extending previous work [6], we derive two new algorithms within the bidirectional composition framework.

First, using a Gauss-Newton procedure with a Forwards update rule, we have proposed the Bidirectional Compositional Lie (BCL) algorithm based on the joint compensation of the image and the template. The BCL algorithm is shown to have local invariance properties with respect to the compensated coordinate frame. Secondly, the PBCL algorithm is an approximation of the BCL which highlights other theoretical properties of the approach. On one hand, in the noise-free case, the PBCL is shown to have similar second-order minimization properties as the ESM approach [5] when considering a projected invariant error. On the other hand, in presence of asymmetric levels of noise, the use of an adaptive projection reduces the amount of noise in the Jacobian matrix and improves the robustness.

Experimental results performed on several images show that the proposed algorithms provide significant improvement of performance in the case of strong noise levels and different levels of noise between the two images. It is almost as performant in terms of frequency of convergence and rate of convergence than the best existing approach in other cases. These properties may be useful for possible applications of this approach in the context of online registration for low-light imagery, or to other cases of strongly corrupted images.

REFERENCES

[1] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981.

[2] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework.," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.
[3] G. Dedeoglu, T. Kanade, and S. Baker, "The asymmetry of image registration and its application to face tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 607–623, May 2007.
[4] S. Benhimane and E. Malis, "Real-time image-based tracking of planes using efficient second-order minimization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, (Sendai, Japan), pp. 943–948, 2004.
[5] S. Benhimane and E. Malis, "Homography-based 2D visual tracking and servoing," *International Journal of Robotics Research*, vol. 26, pp. 661–676, July 2007.
[6] R. Mégret, J.-B. Authesserre, and Y. Berthoumieu, "The bi-directional framework for unifying parametric image alignment approaches," in *European Conference on Computer Vision*, pp. 400–411, 2008.
[7] H.-Y. Shum and R. Szeliski, "Construction of panoramic image mosaics with global and local alignment.," *International Journal of Computer Vision*, vol. 36, no. 2, pp. 101–130, 2000.
[8] T. Vercauteren, X. Pennec, A. Perchant, and N. Ayache, "Diffeomorphic demons: Efficient non-parametric image registration.," *NeuroImage*, vol. 45, no. 2, pp. 561–572, 2009.
[9] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms.," in *IEEE Conference on Computer Vision and Pattern Recognition*, (Kauai, HI, USA), pp. 1090–1097, 2001.
[10] J.-B. Authesserre, R. Mégret, and Y. Berthoumieu, "Asymmetric gradient-based image alignment," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 981–984, April 2009.
[11] Y. Keller and A. Averbuch, "Fast motion estimation using bi-directional gradient methods," *IEEE Transactions on Image Processing*, vol. 13, pp. 1042–1054, August 2004.
[12] E. Bayro-Corrochano and J. Ortegón-Aguilar, "Lie algebra approach for tracking and 3D motion estimation using monocular vision," *Image and Vision Computing*, vol. 25, pp. 907–921, 2007.
[13] R. Brooks and T. Arbel, "Generalizing inverse compositional and ESM image alignment," *International Journal of Computer Vision*, vol. online version, 2009.
[14] P. Cachier, E. Bardinet, D. Dormont, X. Pennec, and N. Ayache, "Iconic feature based nonrigid registration: The PASHA algorithm," *Comp. Vision and Image Understanding*, vol. 89, pp. 272–298, Feb.-March 2003.
[15] H. D. Tagare, D. Groisser, and O. Skrinjar, "Symmetric non-rigid registration: A geometric theory and some numerical techniques," *Journal of Mathematical Imaging and Vision*, vol. 34, no. 1, pp. 61–88, 2009.
[16] A. Bartoli, "Groupwise geometric and photometric direct image registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 2098–2108, December 2008.
[17] G. D. Hager and P. N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 10, pp. 1025–1039, 1998.
[18] F. Alter, Y. Matsushita, and X. Tang, "An intensity similarity measure in low-light conditions," in *European Conference on Computer Vision*, vol. 3954, pp. 267–280, 2006.
[19] S. Lieberknecht, S. Benhimane, P. Meier, and N. Navab, "A dataset and evaluation methodology for template-based tracking algorithms," in *International Symposium on Mixed and Augmented Reality*, pp. 145–151, 2009.

Rémi Mégret (M'09) received the Magistère in computer science and modeling in 2000 from Ecole Normale Supérieure de Lyon, France, and the Ph.D. in Computer Science from INSA de Lyon, France, in 2003. As a Ph.D. student, he was a member of the LIRIS Laboratory, INSA de Lyon, and a visiting student at the LAMP laboratory, University of Maryland. In 2004, he joined the ENSEIRB (Ecole Nationale Suprieure d'Electronique, Informatique et Radiocommunications de Bordeaux) where he teaches signal and image processing. He is currently a member of the Signal and Image Processing Group at IMS Laboratory, IPB / University of Bordeaux, France. His main areas of interest are video analysis, pattern recognition and computer vision.

Jean-Baptiste Authesserre received his Master's degree in engineering from the ENSEIRB (Ecole Nationale Suprieure d'Electronique, Informatique et Tlcommunication de Bordeaux) and the Master's degree in signal and image processing from the university of Bordeaux in 2007. He is currently a Ph.D. student in the Signal and Image Processing Group at IMS Laboratory, ENSEIRB-Matmca / Institut Polytechnique de Bordeaux. His research focuses on image alignment in the context of tracking application.

Yannick Berthoumieu received the Ph.D. in January 1996 in signal processing from the University of Bordeaux I, France. In 1998, he joined the ENSEIRB (Ecole Nationale Suprieure d'Electronique, Informatique et Radiocommunications de Bordeaux) as an Assistant Professor. He is currently a Professor with the Department of Telecommunications, Institut Polytechnique de Bordeaux, and University of Bordeaux. His major research interest includes multidimensional signal analysis, stochastic modelling, texture analysis, and image and video processing. Since 2003, he is with the joint laboratory LASIS between the LAPS department and the Total company within the joint research laboratory IMS UMR 5218.