Automatic estimation of asymmetry for gradient-based alignment of noisy images on Lie group

Jean-Baptiste Authesserre, Rémi Megret, Yannick Berthoumieu

Automatic Estimation of Asymmetry for Gradient-Based Alignment of Noisy Images on Lie Group

Jean-Baptiste Authesserre[a,*], Rémi Mégret[a], Yannick Berthoumieu[a]

[a] *Signal and Image Processing Group, IMS, UMR 5218 CNRS, University of Bordeaux, 351 Cours de la Libération F-33405 TALENCE CEDEX, France. E-mail : jean-baptiste.authesserre@ims-bordeaux.fr, remi.megret@ims-bordeaux.fr, yannick.berthoumieu@ims-bordeaux.fr*

Abstract

Many parametric image alignment approaches assume equality of the images to register up to motion compensation. In presence of noise this assumption does not hold. In particular, for gradient-based approaches, which rely on the optimization of an error functional with gradient descent methods, the performances depend on the amount of noise in each image. We propose in this paper to use the Asymmetric Composition on Lie Groups (ACL) formulation of the alignment problem to improve the robustness in presence of asymmetric levels of noise. The ACL formulation, generalizing state-of-the-art gradient-based image alignment, introduces a parameter to weight the influence of the images during the optimization. Three new methods are presented to estimate this asymmetry parameter: one supervised (MVACL) and two fully automatic (AACL and GACL). Theoretical results and experimental validation show how the new algorithms improve robustness in presence of noise. Finally, we illustrate the interest of the new approaches for object tracking under low-light conditions.

Keywords: Asymmetric image alignment, noisy images, parametric motion estimation, gradient methods, Lie Groups.

1. Introduction

Parametric motion estimation is a fundamental task of many vision applications such as object tracking, image mosaicking, video compression and

augmented reality. To recover the motion parameters, direct image alignment works by optimizing a pixel-based error measure between a moving image and a fixed-image called template. Gradient-based alignment uses more specifically a local optimization relying on the gradients of the error measure with respect to the motion parameters, usually applied in an iterative scheme. These gradients in turn depend on the gradients of the images themselves. Also known as template matching, this method was first introduced for optical-flow computation (Lucas and Kanade, 1981). Since then, many related approaches were proposed. Baker and Matthews (2004) summarized four main classes of algorithms as Forwards Additive, Forwards Compositional, Inverse Additive, and Inverse Compositional.

Mégret et al. (2008) extended this taxonomy of the methods in order to take into account recent approaches: the Efficient Second-order Minimization (ESM) algorithm (Benhimane and Malis, 2007) and the Symmetric Gradient Method (SGM) (Keller and Averbuch, 2004). These methods symmetrically weight the gradients of both images when estimating the update of motion parameters, which yields faster convergence and improved robustness. Another contribution of Benhimane and Malis (2007) was to introduce Lie group parameterization in image alignment, which is a natural and efficient way to parameterize motion (Vercauteren et al., 2009), especially for rigid motion estimation.

The approches using symmetrically weighted gradients have been shown to be more robust (Keller and Averbuch, 2004; Benhimane and Malis, 2007) than the forwards and inverse approaches. Those methods rely implicitely on the assumption that the gradients of both images are equally reliable. In presence of images of different SNR (Signal to Noise Ratio), this assumption does not hold. To improve the robustess of alignment algorithms in that case, Authesserre et al. (2009) introduced the Asymmetric Compositional method as an extension of the ESM approach. This approach is based on the optimization of a closed-form error measure that leads to weighting the gradients of both images asymmetrically according to a free parameter α that tunes the level of asymmetry. Tuning this α parameter produces a family of approaches which can yield better performance than the conventional methods. However the automatic computation of α was not provided.

In this paper, we propose new approaches relying on an Asymmetric Compositional on Lie Group (ACL) formulation of image alignment. The new approaches provide an adaptive computation of α to handle asymmetric levels of noise. The paper is organized as follows :

- In Section 2, we formalize the image alignment problem as a parameter optimization procedure on a Lie Algebra. We introduce the ACL formulation as a generalization of existing algorithms.
- In section 3, we provide a theoretical analysis of the ACL objective function.
  - Its relationship with the ESM (Benhimane and Malis (2007)) optimization procedure and the effect of noise are highlighted.
  - A first algorithm, the MVACL is introduced in the context of a Gauss-Newton optimization of the ACL objective function. This approach focus on the amount of noise only and computes α subject to the knowledge of the relative variance of noise in the two images (section 3.2).
- In section 4, we introduce new algorithms to compute α :
  - The GACL and AACL approaches propose more general heuristics to compute α automatically (sections 4.1 and 4.2).
  - To relieve the overhead for the computation of α, fast approximations of those methods are also proposed (section 4.3).
- In section 5 the state-of-the art and the new approaches performance in the context of image alignment under various noise conditions are evaluated.
- In section 6, we illustrate the interest of the new approaches for object tracking under low-light conditions.

2. Image alignment

2.1. Lie group parameterization of motion

Rigid motion models such as non degenerate affine motion for the Euclidian plane $\mathbf{x} \in \mathbb{R}^2$ and homography for the projective plane $\mathbf{x} \in \mathbb{P}^2$ (Bayro-Corrochano and Ortegón-Aguilar, 2007) can be parameterized using Lie Group. A Lie group $\mathcal{P}$ is a differentiable manifold structured by the composition operation ($\circ$). To any finite dimensional Lie Group is associated a Lie algebra $\mathfrak{P}$ whose underlying finite dimensional vector space is the tangent space to $\mathcal{P}$ at the neutral element $\mathbf{0}$. Locally an increment $\delta\boldsymbol{\mu} \in \mathcal{P}$

around $\mathbf{0}$ can be bijectively reparameterized by an increment $\mathbf{v} \in \mathfrak{P}$ using the exponential map:

$$\delta\boldsymbol{\mu}(\mathbf{v}) = \exp(\mathbf{v}), \tag{1}$$

with the following properties:

$$\exp(-\mathbf{v}) = \exp(\mathbf{v})^{-1}, \tag{2}$$

$$\exp(\alpha\mathbf{v}) \circ \exp(\beta\mathbf{v}) = \exp((\alpha+\beta)\mathbf{v}), \tag{3}$$

for all $\alpha, \beta \in \mathbb{R}$.

With such a mapping we can solve for parameters in the algebra and get the corresponding element of interest $\delta\boldsymbol{\mu}$ using eq. (1). This ensures to obtain $\delta\boldsymbol{\mu}$ inside the group without having to enforce explicit constraints (Vercauteren et al., 2009), as opposed to using the more classic vector space embedding $\mathcal{P} \in \mathbb{R}^p$, where p is the number of parameters to estimate in the vector $\delta\boldsymbol{\mu}$.

In the following, as in (Bayro-Corrochano and Ortegón-Aguilar, 2007; Benhimane and Malis, 2007; Vercauteren et al., 2009), we assume that the parameter space $\mathcal{P}$ forms a Lie group, which acts on image coordinates $\mathbf{x}$ through $\mathbf{W}$. Moreover, we assume that the motion model satisfies group action properties (Miller and Younes, 2001). This action has the following properties, which are related respectively to composition ($\circ$), inversion ($^{-1}$) and parameters of the identity transformation $\mathbf{0}$:

$$\begin{aligned}
\mathbf{W}(\boldsymbol{\mu}^k \circ \delta\boldsymbol{\mu}, \mathbf{x}) &= \mathbf{W}(\boldsymbol{\mu}^k, \mathbf{W}(\delta\boldsymbol{\mu}, \mathbf{x})), & (4)\\
\mathbf{y} = \mathbf{W}(\boldsymbol{\mu}^{-1}, \mathbf{x}) &\Leftrightarrow \mathbf{x} = \mathbf{W}(\boldsymbol{\mu}, \mathbf{y}), & (5)\\
\mathbf{W}(\mathbf{0}, \mathbf{x}) &= \mathbf{x}. & (6)
\end{aligned}$$

Explicit expressions of the group action $\mathbf{W}$ and of the exponential map exp are provided in Appendix B for the case of homography, which will be used in the experiments.

2.2. Asymmetric image alignment on Lie Group

The goal of the algorithms presented in this paper is to align a template image $T(\mathbf{x})$ to an input image $I(\mathbf{x})$, where $\mathbf{x} = (x, y)^T$ is a column vector containing the pixel coordinates. The alignment problem is solved iteratively.

We assume that we know at step k a coarse estimation of the warp parameters $\boldsymbol{\mu}^k$. Aligning two images on Lie group is formalized (Authesserre et al., 2009) as estimating the vector $\mathbf{v} \in \mathfrak{P}$ such that the following discrepancy error is minimal:

$$E_\alpha(\mathbf{v}) = \sum_{i \in 1..N} \left| e_{\alpha,i}(\mathbf{v}) \right|^2 \tag{7}$$

where $e_{\alpha,i}$ represents the error at each pixel $\mathbf{x}_i$ belonging to a region of interest $R = (\mathbf{x}_1, \dots \mathbf{x}_N)$:

$$\begin{aligned} e_{\alpha,i}(\mathbf{v}) =& I(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp((1-\alpha)\mathbf{v}), \mathbf{x}_i)) \\ & - T(\mathbf{W}(\exp(-\alpha\mathbf{v}), \mathbf{x}_i)). \end{aligned} \tag{8}$$

The particularity of eq. (8) is to introduce a tuning parameter $\alpha \in [0, 1]$ corresponding to an asymmetric constraint imposed on the alignment process.

The α coefficient allows us to formulate existing approaches generically: the particular cases $\alpha = 0$ and $\alpha = 1$ correspond to extending the classical approaches (Forwards Compositional (Shum and Szeliski, 2000) and Inverse Compositional (Baker and Matthews, 2004)) to Lie Group parameterization. We call these extensions FCL (Forward Compositional Lie) and ICL (Inverse Compositional Lie). The FCL approach has already been introduced under the name LIEMANIFOLD approach in Guangwei et al. (2008). The case $\alpha = 0.5$ corresponds to the ESM (Efficient Second-order Minimization) algorithm presented by Benhimane and Malis (2007) under conditions discussed in (Mégret et al., 2008; Authesserre et al., 2009). Thus the asymmetric method consists in generalizing previous approaches to any $\alpha \in [0, 1]$. Following this naming scheme we will denote in the sequel ACL (Asymmetric Compositional Lie) the asymmetric approach where α is arbitrarily set a priori.

Aligning two images consists in optimizing locally the error defined by eq. (7) and (8) at step k with respect to $\mathbf{v}$. Once $\mathbf{v}$ is computed, the update rule provides a new estimation of the motion parameters:

$$\boldsymbol{\mu}^{k+1} = \boldsymbol{\mu}^k \circ \exp(\mathbf{v}). \tag{9}$$

Baker and Matthews (2004) showed that the Gauss-Newton (GN) optimization has efficient convergence properties and a reasonable computational

cost. Following most authors (Lucas and Kanade, 1981; Shum and Szeliski, 2000; Keller and Averbuch, 2004; Vercauteren et al., 2009), we will use this approach. In the sequel we denote by $\mathbf{e}_\alpha = (e_{\alpha,i})_{i=1..N}$ the vector obtained by concatenating the pixel-wise errors $e_{\alpha,i}$. The Gauss-Newton optimization of the error function (7) yields:

$$\hat{\mathbf{v}}_\alpha = -(\mathbf{J}_\alpha(\mathbf{0}))^\dagger \ \mathbf{e}_\alpha(\mathbf{0}) = -(\mathbf{J}_\alpha(\mathbf{0})^t \mathbf{J}_\alpha(\mathbf{0}))^{-1} \mathbf{J}_\alpha(\mathbf{0})^t \mathbf{e}_\alpha(\mathbf{0}), \tag{10}$$

where t is the transpose operator and $(\mathbf{J}_\alpha(\mathbf{v}_0))^\dagger$ is the pseudo-inverse of the jacobian matrix of the error vector $\mathbf{e}$ at $\mathbf{0}$. The jacobian matrix is defined as:

$$\mathbf{J}_\alpha(\mathbf{w}) = \left. \frac{\partial \mathbf{e}_\alpha(\mathbf{v})}{\partial \mathbf{v}} \right|_{\mathbf{v}=\mathbf{w}} \tag{11}$$

This matrix can be expressed as the concatenation of the gradients $\mathbf{J}_{\alpha,i}(\mathbf{w})$ of the pixelwise errors $e_{\alpha,i}$: $\mathbf{J}_\alpha(\mathbf{w}) = [\mathbf{J}_{\alpha,1}(\mathbf{w})^t, \dots \mathbf{J}_{\alpha,N}(\mathbf{w})^t]^t$ with

$$\mathbf{J}_{\alpha,i}(\mathbf{w}) = (1-\alpha)\mathbf{J}_{I,i}((1-\alpha)\mathbf{w}) + \alpha \mathbf{J}_{T,i}(-\alpha\mathbf{w}), \tag{12}$$

where $\mathbf{J}_{I,i}$ and $\mathbf{J}_{T,i}$ are the jacobian matrices defined as:

$$\mathbf{J}_{I,i}(\mathbf{w}) = \left. \frac{\partial I(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp(\mathbf{v}), \mathbf{x}_i))}{\partial \mathbf{v}} \right|_{\mathbf{v}=\mathbf{w}}, \tag{13}$$

$$\mathbf{J}_{T,i}(\mathbf{w}) = \left. \frac{\partial T(\mathbf{W}(\exp(\mathbf{v}), \mathbf{x}_i))}{\partial \mathbf{v}} \right|_{\mathbf{v}=\mathbf{w}}. \tag{14}$$

3. Theoretical analysis of the ACL objective function

We now introduce a theoretical study showing the usefulness of the generic ACL formulation to provide new insights into existing gradient-based approaches and the effect of noise. Considering successively the noiseless and the noisy cases, second-order expansion of the error function $\mathbf{e}_\alpha(\mathbf{v})$ is performed. This allows us to highlight in section 3.2 the trade-off that needs to be done between the minimization of both second-order structural terms and noise variance terms.

Let us consider two images corrupted by independant additive noises characterized by respective variances σ_I^2 and σ_T^2:

$$I = I_{nf} + \epsilon_I \quad \text{and} \quad T = T_{nf} + \epsilon_T, \tag{15}$$

where the indice nf indicates noise-free values. This yields the noisy jacobians:

$$\mathbf{J}_I(\mathbf{w}) = \mathbf{J}_I^{nf}(\mathbf{w}) + \mathbf{J}_{\epsilon_I}(\mathbf{w}) \quad \text{and} \quad \mathbf{J}_T(\mathbf{w}) = \mathbf{J}_T^{nf}(\mathbf{w}) + \mathbf{J}_{\epsilon_T}(\mathbf{w}), \tag{16}$$

where $\mathbf{J}_{\epsilon_I}(\mathbf{w})$ and $\mathbf{J}_{\epsilon_T}(\mathbf{w})$ are the jacobian matrices obtained by concatenating the pixelwise gradients:

$$\mathbf{J}_{\epsilon_I,i}(\mathbf{w}) = \left.\frac{\partial \epsilon_I(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp(\mathbf{v}), \mathbf{x}_i))}{\partial \mathbf{v}}\right|_{\mathbf{v}=\mathbf{w}}, \text{ and} \tag{17}$$

$$\mathbf{J}_{\epsilon_T,i}(\mathbf{w}) = \left.\frac{\partial \epsilon_T(\mathbf{W}(\exp(\mathbf{v}), \mathbf{x}_i))}{\partial \mathbf{v}}\right|_{\mathbf{v}=\mathbf{w}}. \tag{18}$$

3.1. Noise-free case

In this section we consider that the noise-free images are identical up to the true motion parameter vector $\mathbf{v}^*$:

$$\forall \mathbf{x} \quad I^{nf}(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp((1-\alpha)\mathbf{v}^*), \mathbf{x}) = T^{nf}(\mathbf{W}(\exp(-\alpha \mathbf{v}^*), \mathbf{x})), \tag{19}$$

which yields:

$$\mathbf{e}_\alpha^{nf}(\mathbf{v}^*) = \mathbf{0}. \tag{20}$$

Under this assumption we can show that:

Theorem 1 (Second-order expansion under noise-free conditions at optimum $\mathbf{v}^*$)**.** For any α:

$$\mathbf{e}_\alpha^{nf}(\mathbf{v}^*) = \mathbf{e}_\alpha^{nf}(\mathbf{0}) + \frac{1}{2}(\mathbf{J}_I^{nf}(\mathbf{0}) + \mathbf{J}_T^{nf}(\mathbf{0}))\mathbf{v}^* + O(||\mathbf{v}^*||^3). \tag{21}$$

Proof. We use here the lemmas presented in Appendix A. According to (A.1), we have to second-order:

$$\mathbf{e}_\alpha^{nf}(\mathbf{v}) = \mathbf{e}_\alpha^{nf}(\mathbf{0}) + \frac{1}{2}(\mathbf{J}_\alpha^{nf}(\mathbf{0}) + \mathbf{J}_\alpha^{nf}(\mathbf{v}))\mathbf{v} + O(||\mathbf{v}||^3). \tag{22}$$

Moreover, for the true parameter $\mathbf{v}^*$, and using (A.13), we have:

$$(\mathbf{J}_\alpha^{nf}(\mathbf{0}) + \mathbf{J}_\alpha^{nf}(\mathbf{v}^*))\mathbf{v}^* = (\mathbf{J}_I^{nf}(\mathbf{0}) + \mathbf{J}_T^{nf}(\mathbf{0}))\mathbf{v}^* + O(||\mathbf{v}^*||^3) \tag{23}$$

Plugging (23) into (22) yields (21). □

Note that $\mathbf{e}_\alpha(\mathbf{0})$ is independant of α. Benhimane and Malis (2007) initially proposed the ESM procedure consisting in plugging the jacobian $\mathbf{J}_{\text{ESM}}$ into the equation (10) as a second-order minimization of the forward objective function ($\alpha = 0$ in (8)) where:

$$\mathbf{J}_{\text{ESM}} = \frac{1}{2}(\mathbf{J}_I(\mathbf{0}) + \mathbf{J}_T(\mathbf{0})). \tag{24}$$

Their proof was based on the assumption (19). Under the same assumption, equation (21) generalizes this result by revealing that the ESM optimization yields in fact a second-order minimization of any ACL objective function, with FCL and ICL objective functions as special cases.

3.2. In presence of noise

Under noisy conditions eq. (23) cannot be applied on the noisy jacobian $\mathbf{J}_\alpha$ and holds only for the noise-free part $\mathbf{J}_\alpha^{nf}$. Furthermore, during the optimization procedure we have only access to the noisy image jacobian $\mathbf{J}_I(\mathbf{0})$ and $\mathbf{J}_T(\mathbf{0})$. Thus, we are looking for the α value that produces a jacobian $\mathbf{J}_\alpha(\mathbf{0})$ as close as possible to the jacobian $\mathbf{J}_{\text{ESM}}^{nf}$.

Theorem 2 (Discrepancy between $\mathbf{J}_\alpha(\mathbf{0})$ and the ideal jacobian $\mathbf{J}_{\text{ESM}}^{nf}$)**.**

$$\mathbf{J}_{\text{ESM}}^{nf} = \mathbf{J}_\alpha(\mathbf{0}) - (\mathbf{A}_\alpha + \mathbf{B}_\alpha), \tag{25}$$

with:

$$\begin{aligned} \mathbf{A}_\alpha &= (\frac{1}{2} - \alpha)\mathbf{J}_\ominus^{nf}(\mathbf{0}), \\ \mathbf{B}_\alpha &= (1-\alpha)\mathbf{J}_{\epsilon_I}(\mathbf{0}) + \alpha\mathbf{J}_{\epsilon_T}(\mathbf{0}), \end{aligned} \tag{26}$$

where :

$$\mathbf{J}_\ominus^{nf}(\mathbf{0}) = \mathbf{J}_I^{nf}(\mathbf{0}) - \mathbf{J}_T^{nf}(\mathbf{0}). \tag{27}$$

Proof.

$$\begin{aligned} \mathbf{J}_{\text{ESM}}^{nf} &= \mathbf{J}_\alpha^{nf}(\mathbf{0}) - (\frac{1}{2} - \alpha)\mathbf{J}_\ominus^{nf}(\mathbf{0}) \\ &= \mathbf{J}_\alpha(\mathbf{0}) - (\frac{1}{2} - \alpha)\mathbf{J}_\ominus^{nf}(\mathbf{0}) - ((1-\alpha)\mathbf{J}_{\epsilon_I}(\mathbf{0}) + \alpha\mathbf{J}_{\epsilon_T}(\mathbf{0})) \end{aligned} \tag{28}$$

□

Equation (25) highlights the presence of two kinds of corrupting terms $\mathbf{A}_\alpha\mathbf{v}^*$ and $\mathbf{B}_\alpha\mathbf{v}^*$ when plugged into eq. (21).

Term of structure. The term $\mathbf{A}_\alpha \mathbf{v}^*$ corresponds to second-order terms of the noise-free error $\mathbf{e}_\alpha^{nf}(\mathbf{v}^*)$ at optimum as it can be shown by plugging (A.13) into (A.4):

$$\mathbf{J}_{\ominus}^{nf}(\mathbf{0})\mathbf{v}^* = (\mathbf{J}_I^{nf}(\mathbf{0}) - \mathbf{J}_I^{nf}(\mathbf{v}^*))\mathbf{v}^* \overset{(A.4)}{=} O(||\mathbf{v}^*||^2). \tag{29}$$

Using the ESM algorithm allows us to compensate them and therefore reach a second-order approximation in the noise-free case.

Term of noise. The term $\mathbf{B}_\alpha \mathbf{v}^*$ corresponds to first-order contribution of noise. Assuming the noises on I and T are independant of respective variance σ_I^2 and σ_T^2 and of zero mean, the total variance of $\mathbf{B}_\alpha \mathbf{v}^*$ is proportional to:

$$\text{var}(\mathbf{B}_\alpha \mathbf{v}^*) \propto (1-\alpha)^2 \sigma_I^2 + \alpha^2 \sigma_T^2 \tag{30}$$

Minimizing (30) with respect to α defines a new Minimal Variance ACL approach denoted MVACL:

$$\alpha^{\text{MVACL}} = \frac{\sigma_I^2}{\sigma_I^2 + \sigma_T^2}. \tag{31}$$

This α value has the following properties: it yields an ESM behaviour for symmetric amount of noise ($\sigma_I = \sigma_T$) and unidirectional FCL behaviour (resp. ICL behaviour) when $\sigma_I \ll \sigma_T$ (resp. $\sigma_I \gg \sigma_T$).

Several typical cases can be considered:

- The SNR is high and the initialization is close to the optimum ($||\mathbf{v}^*||^2 \ll ||\mathbf{v}^*||$): the terms $\mathbf{A}_\alpha$ and $\mathbf{B}_\alpha$ are small and do not influence the alignment process, the FCL, ICL, ESM and MVACL perform equivalently.

- The SNR (signal to noise ratio) is high and the initialization is far from the optimum ($||\mathbf{v}^*||$ is large): the term $\mathbf{A}_\alpha$ dominates $\mathbf{B}_\alpha$ and the optimal α is near 0.5, as used by the ESM.

- The SNR is low, but the noise is symmetrically distributed ($\sigma_I \approx \sigma_T$) on I and T, then the variance of $\mathbf{B}_\alpha \mathbf{v}^*$ is also minimum for $\alpha = 0.5$, which is used by both the ESM and the MVACL algorithms.

- The SNR is low and asymmetrically distributed $\sigma_I \gg \sigma_T$ (resp. $\sigma_I \ll \sigma_T$) and the initialization is close to the optimum : $\mathbf{B}_\alpha$ dominates $\mathbf{A}_\alpha$, and the optimal α is the one that minimizes the variance of $\mathbf{B}_\alpha$ as

	Term of structure $\mathbf{A}_\alpha \mathbf{v}^*$	Term of noise $\mathbf{B}_\alpha \mathbf{v}^*$	Recommended approach
Noiseless or low noise with close initialization	negligible	negligible	FCL, ICL, ESM, MVACL
Noiseless or low noise with far initialization	X	negligible	ESM
Symmetric noise	X	X	MVACL, ESM
Near initialization with strong asymmetric noise	negligible	X	MVACL
Far initialization with strong asymmetric noise	X	X	proposed heuristics AACL and GACL

Table 1: Summary of methods fitness to several typical situations of alignement. Noiseless or low noise : σ_I, σ_T are negligible. Strong noise : σ_I or/and σ_T are high values. Symmetric noise : $\sigma_I \approx \sigma_T$. Near initialization : the motion to estimate $||\mathbf{v}^*||$ is close to 0. Far initialization : the motion to estimate $||\mathbf{v}^*||$ is large.

estimated by the MVACL. For very asymmetric noise levels, one of the images is far more corrupted than the other. The optimal α value can reach the bounds 0 if T is the image strongly corrupted (resp. 1 if I is the image strongly corrupted), which is used by the FCL (resp. ICL) approach.

- The SNR is low and the initialization is far from the optimum : both terms $\mathbf{A}_\alpha$ and $\mathbf{B}_\alpha$ are not negligible and should be taken care of. In that case it is difficult to predict which algorithm is more efficient between the MVACL and the ESM.

It follows from the previous analysis that the ESM and the MVACL provide the same result for symmetrical levels of noise, but that in other cases, each one assumes one of terms $\mathbf{A}_\alpha$ or $\mathbf{B}_\alpha$ is negligible. Those remarks are summarized in table 1 and will be discussed further in the experimental section 5.2. The MVACL furthermore requires the knowledge of the noise variances. We propose in the next section to take a different perspective on this problem by using a geometrically motivated solution for the fully-automatic estimation of α.

4. Fully-automatic estimation of α

Authesserre et al. (2009) have shown that the asymmetric approach can outperform other approaches in terms of robustness to noise by choosing

an appropriate α. In the robotic community Marey and Chaumette (2008) propose a similar approach for designing command law in visual servoing. In both cases no rule was presented to compute the α-value. In this section we propose two methods based on an automatic computation of this coefficient.

*4.1. Method 1 : Geometric ACL (*GACL*)*

In the following, we denote by F_α the affine subspace passing through $\mathbf{e}_\alpha(\mathbf{0})$ and parallel to the span of the columns of $\mathbf{J}_\alpha(\mathbf{0})$. The elements of F_α can be obtained by the linearized error:

$$\tilde{\mathbf{e}}_\alpha(\mathbf{v}) = \mathbf{e}_\alpha(\mathbf{0}) + \mathbf{J}_\alpha(\mathbf{0})\mathbf{v}. \tag{32}$$

We also denote by $\hat{\mathbf{v}}_\alpha$ the Lie algebra increment estimated by plugging $\mathbf{J}_\alpha(\mathbf{0})$ into eq. (10).

The geometric interpretation of the Gauss-Newton algorithm is that the increment $\hat{\mathbf{v}}_\alpha$ is chosen such that $\hat{\mathbf{e}}_\alpha = \tilde{\mathbf{e}}_\alpha(\hat{\mathbf{v}}_\alpha)$ corresponds to the orthogonal projection of the origin $\mathbf{O}$ of the error space onto the affine subspace F_α. Thus, the different ACL methods can be distinguished by the choice of the subspace F_α onto which $\mathbf{O}$ is projected. This is illustrated in fig. 1 for the case of a motion model with one degree of freedom.

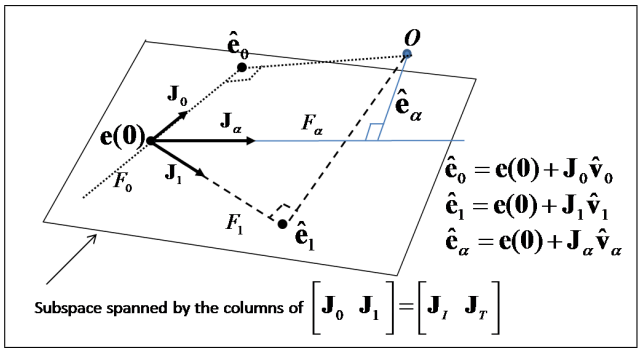

Figure 1: Geometric interpretation of the different approaches using a Gauss-Newton optimization for one parameter motion: $\mathbf{O}$ is projected orthogonally onto a particular affine subspace F_α (in this case, affine line for a model of $p = 1$ parameter). The projection of $\mathbf{O}$ onto F_α is noted $\hat{\mathbf{e}}_\alpha$.

The α value of the ACL approach can be seen as a confidence value on the directions (in the error space), of the subspaces F_0 and F_1. It should favor

the directions that lead closer to the origin $\mathbf{O}$ of the error space. We propose to compute α by using the distances of $\mathbf{O}$ to both subspaces F_0 and F_1. Considering the plane passing through the points $\hat{\mathbf{e}}_0$, $\hat{\mathbf{e}}_1$ and $\mathbf{O}$ (see fig. 2), we denote by $\mathbf{P}$ the orthogonal projection of $\mathbf{O}$ onto the line passing through $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_0$, by l the distance between $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_0$ and by d_1 (respectively d_0) the distance between $\mathbf{O}$ and $\hat{\mathbf{e}}_1$ (respectively $\hat{\mathbf{e}}_0$). Using the Pythagorean theorem in $(\mathbf{O}, \mathbf{P}, \hat{\mathbf{e}}_0)$ and in $(\mathbf{O}, \mathbf{P}, \hat{\mathbf{e}}_1)$, we obtain:

$$\alpha^{\mathrm{GACL}} = \frac{d_0^2 - d_1^2}{2l^2} + \frac{1}{2} = \frac{< \hat{\mathbf{e}}_0 | (\hat{\mathbf{e}}_0 - \hat{\mathbf{e}}_1) >}{||(\hat{\mathbf{e}}_0 - \hat{\mathbf{e}}_1)||^2}, \tag{33}$$

where $< .|. >$ (respectively $||.||$) stands for the regular scalar product (respectively the Euclidean vector norm) in $\mathbb{R}^N$.

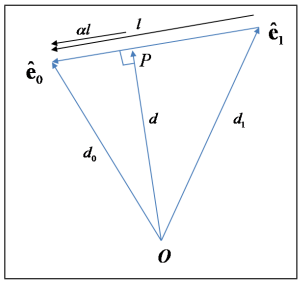

Figure 2: Generic interpretation of the α^{GACL} value: The distances between the point $\mathbf{O}$ and the subspaces F_0 and F_1 are estimated. Then α^{GACL} is chosen to favor the nearest subspace to $\mathbf{O}$.

The α^{GACL} value favors the subspace nearest to $\mathbf{O}$. If both subspaces are equally near to $\mathbf{O}$, the α^{GACL} value is equal to 0.5, and the behaviour of the ESM approach occurs. This method in its simplest form requires computing both $\hat{\mathbf{e}}_0$ and $\hat{\mathbf{e}}_1$ to estimate α^{GACL} which is quite costly.

*4.2. Method 2 : Analytic ACL (*AACL*)*

The AACL algorithm considers the linearization of the pixelwise residual eq. (8) around $\mathbf{v} = \mathbf{0}$ (cf. eq. (32)). An approximation of the complete residual error is obtained by plugging eq. (32) into eq. (7):

$$\begin{aligned} E_\alpha(\mathbf{v}) \approx & ||\mathbf{e}_\alpha(\mathbf{0}) + \mathbf{J}_I(\mathbf{0})\mathbf{v}||^2 \\ & - 2\alpha(\mathbf{e}_\alpha(\mathbf{0}) + \mathbf{J}_I(\mathbf{0})\mathbf{v})^T \mathbf{J}_\ominus(\mathbf{0})\mathbf{v} + \alpha^2 ||\mathbf{J}_\ominus(\mathbf{0})\mathbf{v}||^2, \end{aligned} \tag{34}$$

where $\mathbf{J}_{\ominus} = \mathbf{J}^I(\mathbf{0}) - \mathbf{J}^T(\mathbf{0})$.

Minimizing this residual with respect to α for a fixed $\hat{\mathbf{v}}$ yields the confidence value used by the AACL algorithm:

$$\alpha^{\text{AACL}}(\hat{\mathbf{v}}) = \frac{< \tilde{\mathbf{e}}_0(\hat{\mathbf{v}}) | (\tilde{\mathbf{e}}_0(\hat{\mathbf{v}}) - \tilde{\mathbf{e}}_1(\hat{\mathbf{v}}) >}{||\tilde{\mathbf{e}}_0(\hat{\mathbf{v}}) - \tilde{\mathbf{e}}_1(\hat{\mathbf{v}})||^2}, \tag{35}$$

where $\tilde{\mathbf{e}}_\alpha(\hat{\mathbf{v}})$ is defined by equation (32).

Notice the similarity between the expressions (33) and (35). In the first case, α is obtained by projecting $\mathbf{O}$ orthogonally onto the affine line $(\hat{\mathbf{e}}_0, \hat{\mathbf{e}}_1)$. In the second one, $\mathbf{O}$ is projected orthogonally onto the line $(\tilde{\mathbf{e}}_0(\hat{\mathbf{v}}), \tilde{\mathbf{e}}_1(\hat{\mathbf{v}}))$ instead. For the AACL method, a fixed $\hat{\mathbf{v}}$ has to be set before using eq. (35). In the sequel, we note AACL^M the analytic AACL approach using the method M for computing $\hat{\mathbf{v}}$, where M can be the FCL approach ($\alpha = 0$), the ICL approach ($\alpha = 1$) or the ESM approach ($\alpha = 0.5$).

4.3. Summary of the algorithms and fast approximations

The state-of-the-art gradient-based approaches and the new ones can be seen as particular cases of the ACL formulation using different values of α for the computation of the jacobian matrix eq. (12). The different methods considered in this paper are summarized in the table 2. The main steps of the algorithm scheme is summarized in table 3.

	ACL Methods					
	Non-adaptive to noise			Adaptive to noise		
	Fixed-α			Supervised-α	Automatic-α	
	FCL	ICL	ESM	MVACL	GACL	AACL^M
α	0	1	0.5	α^{MVACL} eq. (31)	α^{GACL} eq. (33)	$\alpha^{\text{AACL}}(\hat{\mathbf{v}})$ eq. (35)

Table 2: The different asymmetric approaches. For the AACL approach, M stands for the method (FCL, ICL or ESM approach) used to set $\hat{\mathbf{v}}$ in equation (35).

For step 2b), if the automatically computed α is such that $\alpha > 1$ (resp. is such that $\alpha < 0$), we arbitrarily set it to 1 (resp. 0) yielding the behaviour of the ICL (resp. FCL) approach. All approaches are based on the computation of the Jacobian matrices $\mathbf{J}_I$ and $\mathbf{J}_T$. Implementation issues are tackled in appendix Appendix C.

As noted by Baker and Matthews (2004), steps 2a, 2b and 2c can be skipped for the ICL approach ($\alpha = 1$). Moreover, for this method, the

Iterative scheme of the ACL **approaches**

1. Precomputation:
   (a) Compute the template jacobian $\mathbf{J}_T(\mathbf{0})$ (eq. (14)),
   (b) Estimate α for the MVACL, F-GACL and F-AACL (eq. (31), (33) and (35)).
2. Iterate until convergence:
   (a) Compute the current error $\mathbf{e}_\alpha(\mathbf{0})$ (eq. (8)) and the image jacobian $\mathbf{J}_I(\mathbf{0})$ (eq. (13)),
   (b) Only for AACL and GACL approaches, compute α (eq. (35) or (33)),
   (c) Compute the asymmetric jacobian $\mathbf{J}_\alpha(\mathbf{0})$ (eq. (12)),
   (d) Solve for $\mathbf{v}$ (eq. (10)),
   (e) Update parameters $\boldsymbol{\mu}^{k+1}$ (eq. (9)).

Table 3: Unified presentation of the algorithm scheme of the ACL approaches.

pseudo-inverse of the jacobian $\mathbf{J}_{\alpha=1}(\mathbf{0}) = \mathbf{J}_T(\mathbf{0})$ can be precomputed. The ICL algorithm is thus the most efficient of the ACL approaches in terms of computational time per iteration.

The AACL and GACL algorithms may introduce a significant overhead in terms of computational time for the computation of α (step 2b) if they are run on a sequential processor. To improve the efficiency, we propose estimating α only once during the first iteration. This yields an approximation of the AACL and GACL denoted F-GACL (Fast-GACL) and F-AACL (Fast-AACL). The fast new approaches only introduce an additional computational effort for computing α during the first iteration compared with fixed α-ACL approaches (except for the ICL approach).

This computational cost is called E-CC (for Extra Computational Cost) in the sequel. Moreover, we denote by P-CC (Pre-Computational Cost) the computational effort induced each time the image used as the template T changes (step 1) in table 3) and by I-CC (iterative computational cost) the standard cost of each iteration of a fixed-α-ACL algorithm (step 2a), 2c) and 2d) in table 3). The table 4 summarizes the computational time of the different approaches obtained with a Matlab implementation on a Intel(R) Core(TM)2 duo CPU 3.0 GHz with 4 GB of RAM. The number of pixels N of the template is 10000. The number of parameters for the homography is $p = 8$.

Methods	P-CC	E-CC	I-CC
FCL	4.22	0	7.00
ICL	7.60	0	4.53
ESM	6.73	0	7.16
AACL^{ESM}	6.38	0	9.08
$F - \text{AACL}^{\text{ESM}}$	6.40	1.33	7.15
GACL	7.57	0	8.99
$F - \text{GACL}$	7.61	1.25	7.15

Table 4: Computational time (in ms) of the different approaches. Three steps are distinguished : the precomputation for the template T (P-CC), the extra computation performed by the fast new approaches each time a new image I has to be processed (E-CC), and the per iteration cost performed by each algorithm (I-CC).

The E-CC for the fast approaches takes approximately a fifth of the computational time of one iteration of the ESM. Thus the additional computational cost for computing α with the fast approximations of the GACL and AACL is very small when several iterations are required for convergence. The next section will show in which situations this cost is counterbalanced by an increased robustness of the estimation.

5. Experimental validation

The performance assessment uses a methodology inspired by the benchmark proposed by Baker and Matthews (2004). Random homographies are generated by adding spatial Gaussian noise of standard deviation σ_p (called Point Sigma) to four canonical point locations in a reference image I_{ref}. The new locations are called test points. Using these homographies, the image I_{ref} is warped onto the template image T_{ref}. I_{ref} (resp. T_{ref}) is then corrupted with additive gaussian noise: $I = I_{ref} + \epsilon_I$ (resp. $T = T_{ref} + \epsilon_T$). Noise level is characterized by the SNR (Signal to Noise ratio) between the noise-free image and the image of noise: $SNR_{dB} = 10 \log_{10} \left(\frac{E(I^2)}{Var(\epsilon)} \right)$. The RMS point error is defined as the root mean square error between the test points and the point locations obtained from the motion parameter estimated by an alignment algorithm in the reference image coordinates.

We will consider two main performance criteria defined by Baker and Matthews (2004):

- Average frequency of convergence: percentage of tests where an algorithm converged to the correct estimate (RMS Point Error < 1 pixel),

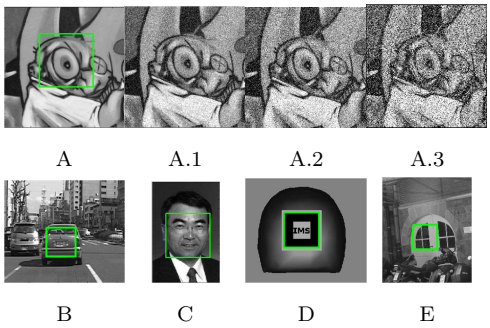

Figure 3: Images used for the experiments. First raw: The noise-free image (left) is used for extracting the template image. Image I is then obtained by adding gaussian noise to the noise free image. The noisy images (right) are shown with respective SNR : 15 dB, 10 dB and 5 dB. Second raw: other images 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 Matthews (2004). Image D was synthetized using an openGL implementation.

- 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 sequel, the results are obtained by averaging the performances obtained on the five images [A-E] shown in figure 3. The motion model is a homography parameterized as in Appendix B: a 3×3 homography matrix $\mathbf{H}$ which belongs to the Lie group $\mathcal{P}$ of matrices with $\det(\mathbf{H}) = 1$. Its associated Lie algebra $\mathfrak{P}$ is the vector space of matrices with null trace. 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 which converge are averaged per image.

5.1. ACL *algorithms behavior in presence of noise*

Figure 4 shows the behaviours of the ACL approaches, using a fixed value for α, in terms of frequency of convergence with respect to image SNR, when aligning an almost noise free template to a noisy image. As mentioned by Authesserre et al. (2009), the best performance is obtained by a compromise between the use of the gradients of both T and I, and the weighting of the influence of noise. Thus, when the images are noise free (high SNR), the FCL ($\alpha = 0$) and the ICL ($\alpha = 1$) provide identical results and the ESM ($\alpha = 0.5$) outperforms these two methods. However, when the noise level increases on I, the performance of the ESM decreases. When the SNR of I

becomes too low, the ICL provides the best results. Between these two cases the best robustness is obtained using an intermediate α value.

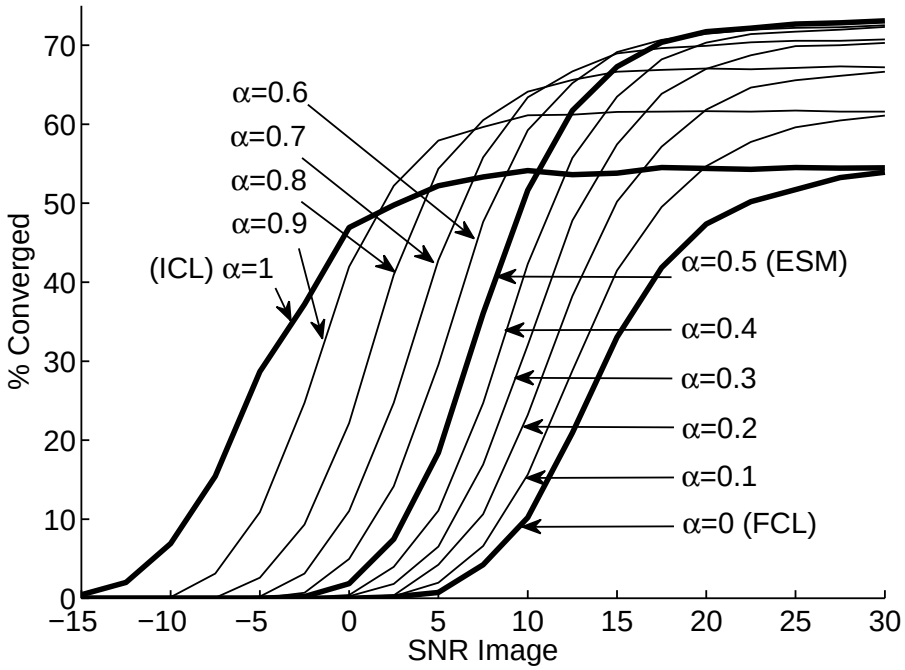

Figure 4: Average frequency of convergence of fixed-α ACL approaches with respect to image SNR for a Point Sigma equal to 12 pixels. A SNR of 100 dB was used on the template. The image is corrupted using a SNR varying from -15 dB to 30 dB.

5.2. Influence of the noise asymmetry

To simulate controled noise asymmetry, noise is parameterized by a total variance σ^2, and an asymmetry coefficient $\beta \in [0,1]$. The total variance σ^2 is computed from a given global SNR in order to average results with images (figure 3) of the same quality. The amount of noise is allocated to I and T as follows:

$$\sigma_I^2 = (1-\beta)\sigma^2,\ \sigma_T^2 = \beta\sigma^2,\ \text{with } \beta \in [0,1]. \tag{36}$$

where σ_I (resp. σ_T) is used as noise standard deviation to corrupt the image I (resp. the template T) with an additive centered white Gaussian noise.

Six configurations are considered to evaluate the comparative performances for various typical situations:

- Exp.0 : near initialization (Point Sigma σ_p equal to 6 pixels) and high SNR (Total SNR=15 dB)
- Exp.1 : near initialization (Point Sigma σ_p equal to 6 pixels) and medium SNR (Total SNR=10 dB)
- Exp.2 : near initialization (Point Sigma σ_p equal to 6 pixels) and low SNR (Total SNR=5 dB)
- Exp.3 : far initialization (Point Sigma σ_p equal to 12 pixels) and high SNR (Total SNR=15 dB)
- Exp.4 : far initialization (Point Sigma σ_p equal to 12 pixels) and medium SNR (Total SNR=10 dB)
- Exp.5 : far initialization (Point Sigma σ_p equal to 12 pixels) and low SNR (Total SNR=5 dB)

The results of the experiments 0-5 are shown on figure 5 and discussed in the sequel.

5.2.1. Conventional algorithms

The first column of the figure 5 shows the results of the FCL, ICL, ESM and MVACL algorithms. As mentioned in the theoretical analysis (section 3.1), the ESM provides better results when β is near 0.5 (see exp. 0-5), or when the total SNR is high (Exp. 0 and 3). By using only the gradients of I (resp. T), the FCL (resp. ICL) yields only the best performance for β near 1 (resp. 0) or when the image T (resp. I) is sufficiently corrupted (Exp. 2, low SNR).

5.2.2. MVACL *approach*

As it can be seen on the first column of the figure 5, the MVACL provides performance similar to ICL for $\beta = 0$ that evolves smoothly to be similar to ESM for $\beta = 0.5$ and to FCL for $\beta = 1$. On the one hand, the MVACL yields a better behaviour than the standard approaches under low SNR conditions (Exp.2 and 5) because in that case the MVACL appropriately averages the term of noise $\mathbf{B}_\alpha$ of the equation (25). On the other hand, this approach

is outperformed by the ESM under high SNR conditions (Exp. 0 and 3). Indeed, by only taking into account the noise variance for computing α, the MVACL optimally weights noise-jacobian terms at the expense of taking into consideration the term $\mathbf{A}_\alpha$, which is in that case not negligible compared to the term of noise $\mathbf{B}_\alpha$.

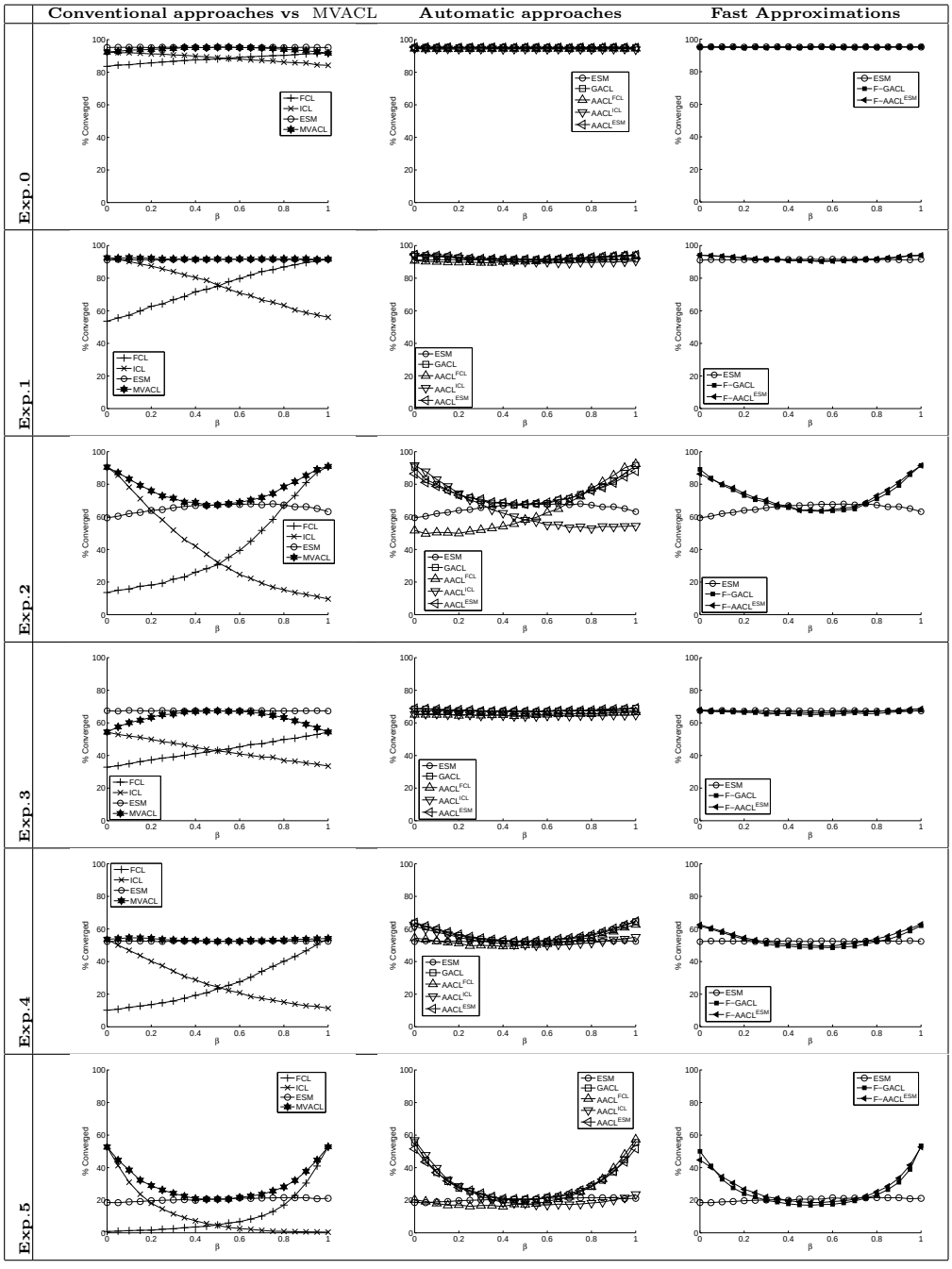

Figure 5: Average frequency of convergence with respect to noise asymmetry for the different approaches. Exp. 0: $\sigma_p = 6$ pixels, SNR=15 dB. Exp. 1: $\sigma_p = 6$ pixels, SNR=10 dB. Exp. 2: $\sigma_p = 6$ pixels, SNR=5 dB. Exp. 3: $\sigma_p = 12$ pixels, SNR=15 dB. Exp. 4: $\sigma_p = 12$ pixels, SNR=10 dB. Exp. 5: $\sigma_p = 12$ pixels, SNR=5 dB. Corresponding numerical values for characteristic β values are presented in table 5.

Conditions	**Exp.0** $\sigma_p = 6$ pixels SNR=15 dB			**Exp.1** $\sigma_p = 6$ pixels SNR=10 dB			**Exp.2** $\sigma_p = 6$ pixels SNR=5 dB			**Exp.3** $\sigma_p = 12$ pixels SNR=15 dB			**Exp.4** $\sigma_p = 12$ pixels SNR=10 dB			**Exp.5** $\sigma_p = 12$ pixels SNR=5 dB		
β	0	0.2	0.5	0	0.2	0.5	0	0.2	0.5	0	0.2	0.5	0	0.2	0.5	0	0.2	0.5
FCL	83.6	85.7	88.2	53.6	62.7	75.0	13.6	18.2	30.6	32.8	37.3	43.2	10.2	13.6	23.3	0.8	1.6	4.8
ICL	92.4	91.2	88.9	92.4	87.4	75.7	90.4	63.6	32.0	54.4	49.9	42.8	53.6	40.2	24.4	52.7	18.2	4.5
ESM	95.2	95.0	95.4	91.0	91.3	**91.7**	59.4	63.9	67.3	67.4	67.2	**67.4**	52.2	52.4	52.3	18.6	19.7	**20.7**
MVACL	92.4	94.0	**95.4**	92.4	92.2	**91.7**	90.4	**76.0**	67.3	54.4	63.4	**67.4**	53.6	54.2	52.3	52.7	**29.0**	**20.7**
GACL	95.1	95.0	95.1	**94.4**	92.5	91.4	90.5	73.3	67.6	68.5	67.4	66.4	63.6	56.0	52.2	55.4	27.5	20.5
AACL$^{\text{FCL}}$	94.5	94.3	94.6	90.7	89.7	90.2	51.6	49.9	58.2	65.1	64.4	64.5	54.4	51.2	50.1	20.3	17.1	17.4
AACL$^{\text{ICL}}$	94.4	94.2	94.3	93.7	92.4	90.0	**91.6**	74.3	58.1	65.3	65.1	64.0	61.5	54.8	50.3	**56.8**	28.6	18.0
AACL$^{\text{ESM}}$	**95.3**	**95.1**	95.2	**94.4**	**92.9**	91.6	86.4	73.3	**67.8**	**68.9**	**68.0**	67.1	**63.9**	**56.6**	**52.6**	51.5	28.0	20.6
F-GACL	95.0	94.8	95.0	94.1	92.4	90.6	89.1	73.0	63.6	67.4	66.4	65.1	61.8	53.7	48.7	50.0	24.0	17.0
F-AACL$^{\text{ESM}}$	95.2	94.8	95.1	94.0	92.5	90.6	86.3	74.4	64.2	68.0	67.0	66.1	62.6	54.6	50.1	44.7	26.7	18.6

Table 5: Average frequency of convergence with respect to noise for the different approaches for characteristic β values. Best overall method is in bold, best methods in each category are underlined.

5.2.3. Influence of the choice of $\hat{\mathbf{v}}$ for the AACL *approaches*

In the following we evaluate the new approaches. The second column of figure 5 shows the different behaviors of the AACL$^{\text{FCL}}$, AACL$^{\text{ICL}}$, AACL$^{\text{ESM}}$ and the GACL algorithms in terms of frequency of convergence with respect to image noise asymmetry. The ESM performances are also plotted as reference. We observe that:

- For any Point Sigma and Total SNR and for any β, the AACL^{M} approach using the method M to initialize $\hat{\mathbf{v}}$ in (35) always provides at least as good results as the method M itself (Exp.0-5).

- The performance of the AACL^{M} is correlated to the performance of the method M itself. Thus, for situations where the FCL (resp. ICL) works well, such as in Exp.2 and and in Exp.5 for β near 1 (resp. near 0), the AACL$^{\text{FCL}}$ provides better results than the conventional methods and other AACL approaches by improving the performances of the FCL (resp. ICL) approach. In all other cases, the AACL$^{\text{ESM}}$ provides better results.

5.2.4. AACL *vs* GACL

For any β value and for any global SNR, the GACL and AACL$^{\text{ESM}}$ yield similar results. Moreover, they always provide the best results (Exp. 0, 1, 3 and 4) or near the best results (Exp. 2 and 5). The AACL$^{\text{FCL}}$ and AACL$^{\text{ICL}}$ are only more robust in very asymmetric cases (Exp.2 and 5) where the noisy image is not the one used for computing the gradients allowing the estimation of $\hat{\mathbf{v}}$. For all these reasons, we consider in the sequel only the GACL and AACL$^{\text{ESM}}$ approaches.

5.3. Performance of the fast approximations

The figure (5), third column shows the different behaviors of the ESM, F-AACL$^{\text{ESM}}$ and F-GACL algorithms in terms of frequency of convergence with respect to image noise asymmetry. It can be seen that the fast approximations produce the same behaviors as the exact AACL and GACL approaches. However, the fast approximations provide slightly worse results than the exact approaches, particularly for quasi symmetric amount of noise (β between 0.35 and 0.65) where they are outperformed by the ESM approach (see Exp. 2, 4 and 5). The fast approaches are thus a good choice for aligning a noisy image to another noisy image, but of greater quality (which

in practice takes place for example in image mosaicking (Pham et al., 2005) where a noisy image is aligned to the current mosaic).

To illustrate this, fig. 6 shows the average rate of convergence (Point Sigma σ_p equal to 6 pixels) and average frequency of convergence with respect to σ_p for a noisy image ($SNR_I = 10$ dB) and a high quality template ($SNR_T = 100$ dB). For legibility, only the fast approximations and the state-of-the art approaches are plotted.

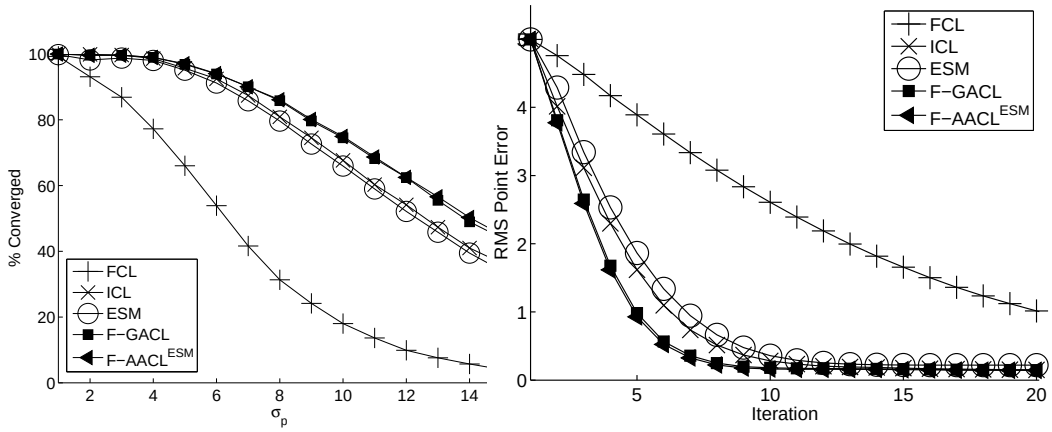

Figure 6: Average Frequency of convergence and average Convergence Rate for the ESM, ICL, FCL, $F-\text{AACL}^{\text{ESM}}$, $F-\text{GACL}$ algorithms in presence of image noise. A SNR of 10 dB has been used on the image and 100 dB on the template.

The fast new approaches outperform the state-of-the-art approaches in terms of frequency of convergence and convergence rate. The improvement in convergence rate counter-balances the E-CC introduced by the computation of α during the first iteration. Thus, the new approaches are at least as efficient as the best reference algorithms while better handling noisy images. Thus, for an equivalent computational complexity and according to the repartition of noise on images we can provide the following recommandation:

- In quasi noiseless condition or in symmetric noise condition ($SNR_I \approx SNR_T$), the ESM algorithm would provide the best results among the studied approaches.

- In very asymmetric noise condition $SNR_I \gg SNR_T$ (resp. $SNR_I \ll SNR_T$), the FCL algorithm (resp. ICL) would provide the best results among the studied approaches.

- In the other cases, the AACL$^{\text{ESM}}$ and GACL would be prefered if the best robustness is required. Trading off for computational performance, the F-AACL$^{\text{ESM}}$ or F-GACL would be chosen in presence of strong corruption with asymmetric noise levels.

6. Object tracking under low-light conditions

In low-light conditions, an optical imaging system produces bad quality images (cf. figure 7) that can be modeled according to Alter et al. (2006) : 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 for 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.

6.1. Corpus and ground truth design

We considered two sequences of 40 and 32 images captured with the same imaging device (samples are presenred on figure 7). In the first sequence one object of interest is selected and three are selected in the second sequence. This yields 136 independent observations of real shot noise associated to low-light imaging. For each sequence and object, a ground-truth is generated by using the following protocol:

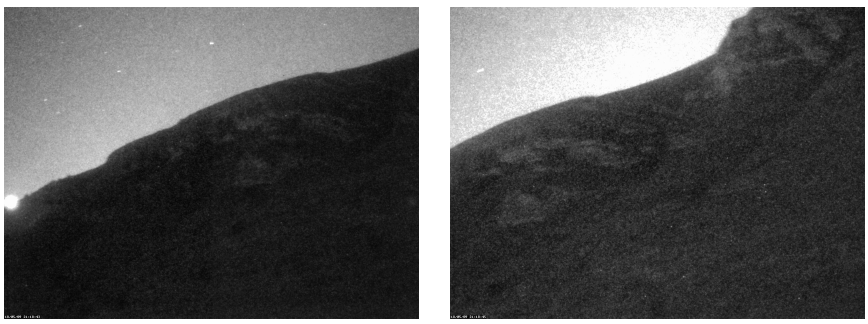

Figure 7: Samples of low-light image.

- the object of interest is selected in the frame 1 of the sequence (which plays the role of the template image T) through the use of a rectangular bounding-box. This location defines the true parameter $\bar{\boldsymbol{\mu}}(1)$ of the position of the object in the first image.

- The ESM algorithm is run to align image $t+1$ and image 1 by initializing alignment parameter with $\bar{\boldsymbol{\mu}}(t)$ and by using 200 iterations which allows the algorithm to estimate parameter $\bar{\boldsymbol{\mu}}(t+1)$. The ESM algorithm is used here because it has been shown to be the more efficient in presence of symmetric noise. The motion model used is 2D translation. We checked manually frame by frame that the ESM produced reasonable motion parameters.

The generated ground-truth $\bar{\boldsymbol{\mu}}$ is approximative. We will not use it to do accuracy evaluation. However, we can use it to evaluate convergence rate and frequency of convergence of the different algorithms with a sufficiently large error threshold.

6.2. Evaluation methodology

The following results are obtained by averaging performance obtained for many different motion initializations for each image and for each object of interest. In order to do this, we use a benchmark similar to the one presented in section 5. First, the corners associated to the object of interest in the current image t are computed by using the parameters $\bar{\boldsymbol{\mu}}(t)$. A perturbation $\delta\mathbf{x}$ drawn randomly from a zero-mean normal distribution of standard deviation $\sigma_p = 10$ is added to the spatial location $\mathbf{x}$ of each corner. The new locations of the corners yield the initial parameters $\boldsymbol{\mu}_I^0(t)$ used to initialize the different alignment algorithms.

Two setups are studied here :

- Exp.6 : the algorithms are run using only the first image as a template. The level of noise is thus symmetric.

- Exp.7 : each algorithm generates its template by averaging the five first frames of the video, after compensating the images in the same coordinate frame by using the estimated motion parameters.

For each frame, 20 different motion initializations are generated which yield more than 2000 tests for each setup.

6.3. Results

Figures 8 show the performance of the different algorithms in terms of mean RMS Point Error and frequency of convergence as a function of the iteration number for the Exp.6 and 7. The mean RMS Point Error corresponds to the mean euclidean distance between the four corners of the region of interest predicted by the tested algorithm and those of the ground truth. For the frequency of convergence, the convergence threshold is 4 pixels in terms of RMS Point Error.

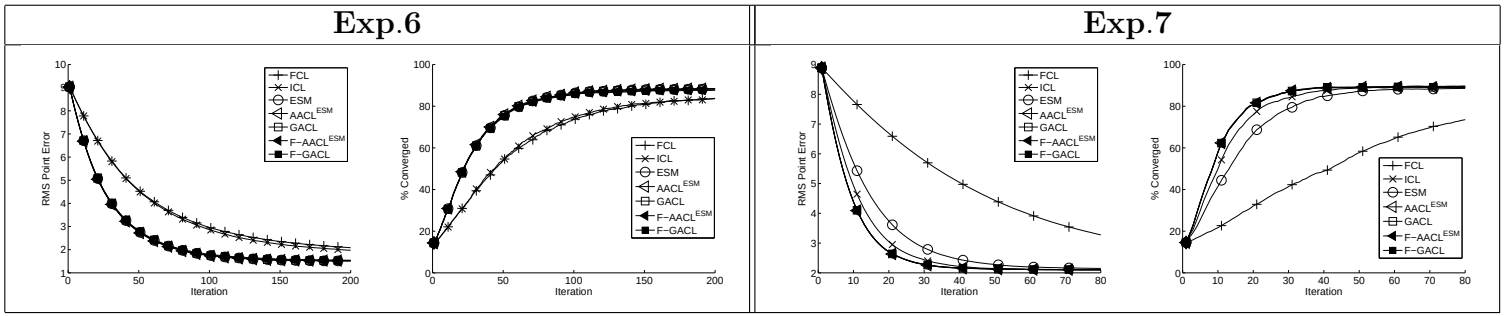

Figure 8: Convergence rate in terms of mean RMS Point error relatively to the iteration number and frequency of convergence relatively to the iteration number on a sequence of real images corrupted by shot noise. A threshold of 4 pixels on the RMS Point Error is used to check the convergence of an approach. Left : only the first frame is used as a template (Exp.6). Right : The five first frames are used to generate the template after motion compensation (Exp.7).

The ESM algorithm and the new approaches ACL provide better results than the more conventional approaches FCL and ICL for the Exp.6 which confirms on real data the good behavior of such approaches under symmetrical level of noise.

The use of an averaged template (Exp.7) improves the performance of all approaches that use the gradients of T. This improvement yields the approaches AACL, GACL and their fast approximations to outperform the other approaches in terms of convergence rate and frequency of convergence.

Remark on the accuracy of the approaches. In figure 8, after convergence, using only the first frame as the template (Exp. 6) has lower RMS point error than generating the template with the first five frames (Exp.7). The difference in accuracy observed under symmetric or asymmetric conditions are around 0.5 pixel. Because the images are very corrupted, it is difficult to check manually that the ground truth provides the position of the object with such a precision. Thus, there are two possible explanations for the difference in RMS Point Error at convergence :

1. By reducing the noise injected to the system in the Exp.7, averaging yields results which are more accurate than without the averaging (inaccurate ground-truth).
2. Because of a non perfect motion estimation, frame averaging in Exp.7 introduces a slight blur on the template which yields a decrease of the accuracy of the approaches (blurred template).

This analysis reveals that, when averaging frames, a trade off has to be made between noise and blur reduction.

The averaging process was only introduced to illustrate the potential of the ACL formulation to register images in presence of noise level asymmetry. The study of using many frames to generate a high quality template is an interesting perspective, which is out of the scope of this paper.

In both Exp.6 and Exp.7 the automatic-ACL approaches provide as accurate results as the best of the state-of-the art approach, while improving robustness and convergence rate under asymmetric conditions, thus showing their ability to estimate a relevant α on such real datasets.

6.4. Illustration of tracking

To illustrate the principle of tracking using an average template, figure 9 presents the results of the approaches FCL, ICL, ESM and F-AACL$^{\mathrm{ESM}}$ (the

different heuristics AACL, GACL, F-AACL$^{\mathrm{ESM}}$ and F-GACL$^{\mathrm{ESM}}$ approaches provide similar results on those data) under real tracking conditions. As in Exp.7, each algorithm generates its template by averaging the five first frames of the video, after compensating the images in the same coordinate frame by using the estimated motion parameters. The motion parameters are initialized in each frame by those estimated in the previous frame. Each algorithm performs 30 iterations per image. The FCL approach fails to track the object of interest. The approach ESM looses the object of interest after the frame 20. The ICL approach accumulates tracking error and fails to track after the frame 25. The F-AACL$^{\mathrm{ESM}}$ approach is able to track the object during the whole sequence.

Such an illustration shows under a realist context, the potentiality of the new Asymmetric approaches.

7. Conclusion

In this paper we used the Asymmetric Composition on Lie Group (ACL) formulation proposed by Authesserre et al. (2009) to provide new gradient-based image alignment methods that compute automatically the asymmetry parameter α. A theoretical analysis allowed us to identify that in presence of noise, the role of the asymmetry parameter α is to perform a trade-off involving two terms : the deterministic structural term and the random noise term. Three new approaches have been proposed, i.e. the MVACL, GACL and AACL. The MVACL approach is more of a theoretical interest as it ignores the structural term completely, in favor of the minimization of noise variance. The two other approaches GACL and AACL are heuristics based on a geometrical and an analytical interpretation of the problem, which were shown to estimate relevant values of α, yielding improvement in the robustness to asymmetric noise. Those heuristics have practical interest, since they are fully automatic and reach comparable performance as the MVACL for asymmetric noise, even though the later requires the knowledge of noise variances. An interesting perspective for a future work, would be to take into account this information to further improve the performances.

The new methods introduce a computational overhead because of the need to compute α. To alleviate the computational burden, fast approximations of the new methods have been introduced, which have comparable computational complexity as fixed-α state-of-the-art methods, but with better robustness in case of asymmetric noise levels.

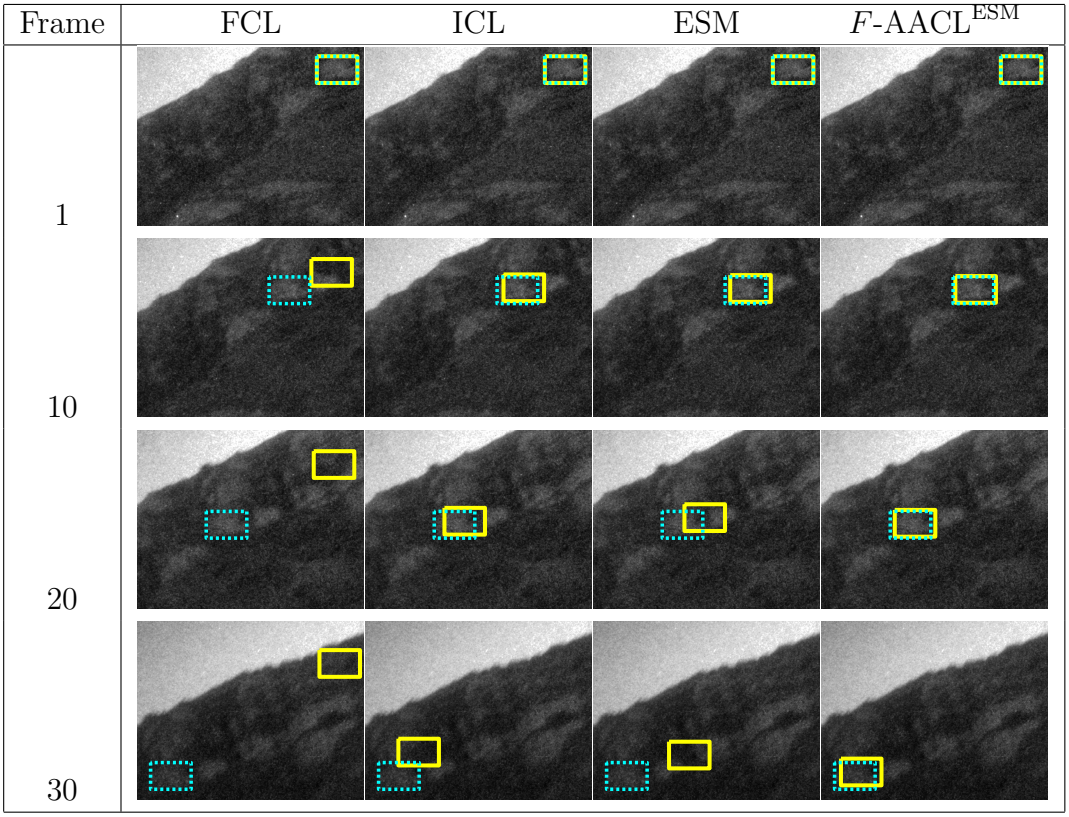

Figure 9: Tracking performance of the FCL, ICL, ESM and F-AACL$^{\text{ESM}}$ approaches on a video captured under low-light conditions. The first five images are used to build an average template. The dashed line rectangle corresponds to the ground truth location of the object. The solid line rectangle corresponds to the tracking results of the different algorithms after 30 iterations.

The benefits of the new methods were studied in the case of image alignment with L2 pixelwise errors. As done by Brooks and Arbel (2009) for the inverse compositional and the ESM methods, the proposed approach could be extended to more generic error metrics and motion models to fit particular needs in various applications. We think their properties could be useful in applications where a slighty noisy template would be aligned to a highly noisy image under low-light condition (Alter et al., 2006).

In this paper, we focused on the problem of noise related asymmetry. In applications such as super-resolution (Dijk et al., 2008), where the images to align can have different resolutions, the derivatives of the low-resolution frame is less reliable than the derivatives of the high resolution estimate. This problem of resolution asymmetry has been thoroughly discussed in (Dedeoglu et al., 2007) in term of bias on the estimation of the warp. However, the problem of the reliability of the image derivatives has not been tackled. We think that it is an interesting perspective for future developments, to merge those two sources of asymmetry in a single formulation.

Aknowledgement

The authors would like to thank M. Grossetete of Thales Avionics corp. for providing the real low light images used in the experimental section.

Appendix A. Second-order approximation of the ACL objective function

In the following we present the main properties used for the derivation of the second-order approximation of the ACL objective function (cf. eq. (21)) under noise-free conditions.

Lemma (Second-order linearization of the ACL objective function)**.**

$$\mathbf{e}_\alpha(\mathbf{v}) = \mathbf{e}_\alpha(\mathbf{0}) + \frac{1}{2}(\mathbf{J}_\alpha(\mathbf{0}) + \mathbf{J}_\alpha(\mathbf{v}))\mathbf{v} + O(||\mathbf{v}||^3) \tag{A.1}$$

Proof. By developping (8) to second-order around $\mathbf{v} = \mathbf{0}$, we have at each pixel $\mathbf{x}_i$:

$$\mathbf{e}_{\alpha,i}(\mathbf{v}) = \mathbf{e}_{\alpha,i}(\mathbf{0}) + \mathbf{J}_{\alpha,i}(\mathbf{0})\mathbf{v} + \frac{1}{2}\mathbf{v}^t\mathbf{H}_{\alpha,i}(\mathbf{0})\mathbf{v} + O(||\mathbf{v}||^3) \tag{A.2}$$

with $\mathbf{J}_{\alpha,i}(\mathbf{0})$ the jacobian matrix and $\mathbf{H}_{\alpha,i}(\mathbf{0})$ the Hessian matrix of $e_{\alpha,i}$ at $\mathbf{0}$:

$$\mathbf{H}_{\alpha,i}(\mathbf{w}) = \left.\frac{\partial^2 \mathbf{e}_{\alpha,i}(\mathbf{v})}{\partial \mathbf{v}^2}\right|_{\mathbf{w}} \tag{A.3}$$

608 We also have:

$$\mathbf{J}_{\alpha,i}(\mathbf{v}) = \mathbf{J}_{\alpha,i}(\mathbf{0}) + \mathbf{v}^t \mathbf{H}_{\alpha,i}(\mathbf{0}) + O(||\mathbf{v}||^2) \tag{A.4}$$

609 Thus, by plugging (A.4) into (A.2), we obtain (A.1). This linearization is
610 valid in noise-free as well as in noisy cases. □

Lemma (Jacobian equivalence along $\mathbf{v}^*$)**.**

$$\mathbf{J}_I^{nf}((1-\alpha)\mathbf{v}^*)\mathbf{v}^* = \mathbf{J}_T^{nf}(-\alpha\mathbf{v}^*)\mathbf{v}^* \tag{A.5}$$

611 *Proof.* Using directional derivative along $\mathbf{v}^*$, the assumption (20) and the
612 exponential map properties (2) and (3), we have:

$$\begin{aligned}
&\mathbf{J}_{I,i}^{nf}((1-\alpha)\mathbf{v}^*)\mathbf{v}^* \\
&\qquad = \left.\frac{\partial I^{nf}(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp((1-\alpha)\mathbf{v}^* + t\mathbf{v}^*), \mathbf{x}_i))}{\partial t}\right|_{t=0} \\
&\qquad = \left.\frac{\partial T^{nf}(\mathbf{W}(\exp(-\alpha\mathbf{v}^* + t\mathbf{v}^*), \mathbf{x}_i))}{\partial t}\right|_{t=0} \\
&\qquad = \mathbf{J}_{T,i}^{nf}(-\alpha\mathbf{v}^*) \left.\frac{\partial(-\alpha\mathbf{v}^* + t\mathbf{v}^*)}{\partial t}\right|_{t=0} \\
&\qquad = \mathbf{J}_T^{nf}(-\alpha\mathbf{v}^*)\mathbf{v}^*
\end{aligned} \tag{A.6}$$

613 □

Corollary.

$$\mathbf{J}_\alpha^{nf}(\mathbf{v}^*)\mathbf{v}^* = \mathbf{J}_I^{nf}((1-\alpha)\mathbf{v}^*)\mathbf{v}^* = \mathbf{J}_T^{nf}(-\alpha\mathbf{v}^*)\mathbf{v}^* \tag{A.7}$$

614

615 with the special cases:

$$\mathbf{J}_I^{nf}(\mathbf{v}^*)\mathbf{v}^* = \mathbf{J}_T^{nf}(\mathbf{0})\mathbf{v}^* \tag{A.8}$$

$$\mathbf{J}_T^{nf}(-\mathbf{v}^*)\mathbf{v}^* = \mathbf{J}_I^{nf}(\mathbf{0})\mathbf{v}^* \tag{A.9}$$

Proof. We have:

$$\mathbf{J}_{\alpha}^{nf}(\mathbf{v}^*)\mathbf{v}^* = (1-\alpha)\mathbf{J}_{I}^{nf}((1-\alpha)\mathbf{v}^*)\mathbf{v}^* + \alpha\mathbf{J}_{T}^{nf}(-\alpha\mathbf{v}^*)\mathbf{v}^*. \tag{A.10}$$

Using lemma A.5 into the previous equation to replace $\mathbf{J}_{nf}^{T}(-\alpha\mathbf{v}^*)\mathbf{v}^*$ yields (A.7). The special cases are obtained for $\alpha = 0$ and $\alpha = 1$. □

Note that we generalize here the proof of (A.8) proposed by Benhimane and Malis (2007) to the generic ACL jacobian.

Lemma (Linear approximation of the image jacobian)**.**

For $\gamma \in [0,\ 1]$

$$\mathbf{J}_I(\gamma\mathbf{v}) = \mathbf{J}_I(\mathbf{0}) + \gamma(\mathbf{J}_I(\mathbf{v}) - \mathbf{J}_I(\mathbf{0})) + O(||\mathbf{v}||^2) \tag{A.11}$$

Proof. By expanding $\mathbf{J}_{I,i}(\gamma\mathbf{v})$ to first-order around $\mathbf{0}$ we have:

$$\mathbf{J}_{I,i}(\gamma\mathbf{v}) - \mathbf{J}_{I,i}(\mathbf{0}) = \gamma\mathbf{v}^t \left.\frac{\partial^2 I(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp(\mathbf{v}), x_i))}{\partial\mathbf{v}^2}\right|_{\mathbf{v}=\mathbf{0}} + O(||\gamma\mathbf{v}||^2) \tag{A.12}$$

□

Lemma (Linear approximation of the α-ACL jacobian)**.**

$$\mathbf{J}_{\alpha}^{nf}(\mathbf{v}^*)\mathbf{v}^* = (\alpha\mathbf{J}_{I}^{nf}(\mathbf{0}) + (1-\alpha)\mathbf{J}_{T}^{nf}(\mathbf{0}))\mathbf{v}^* + O(||\mathbf{v}^*||^3) \tag{A.13}$$

Proof.

$$\begin{aligned}\mathbf{J}_{\alpha}^{nf}(\mathbf{v}^*)\mathbf{v}^* &\overset{(A.7)}{=} \mathbf{J}_{I}^{nf}((1-\alpha)\mathbf{v}^*)\mathbf{v}^* \\ &\overset{(A.11)}{=} (\mathbf{J}_{I}^{nf}(\mathbf{0}) + (1-\alpha)(\mathbf{J}_{I}^{nf}(\mathbf{v}^*) - \mathbf{J}_{I}^{nf}(\mathbf{0})))\mathbf{v}^* + O(||\mathbf{v}^*||^3) \\ &\overset{(A.8)}{=} (\alpha\mathbf{J}_{I}^{nf}(\mathbf{0}) + (1-\alpha)\mathbf{J}_{T}^{nf}(\mathbf{0}))\mathbf{v}^* + O(||\mathbf{v}^*||^3)\end{aligned} \tag{A.14}$$

□

Appendix B. Parametrization of homography

In this section we present the homography parameterization of motion used in the experimental section.

A 2D Homography is a transformation in the projective image plane $\mathbb{P}^2$. A homography can be represented by a 3×3 matrix $\mathbf{H}$:

$$\mathbf{H} = \begin{pmatrix} 1+\mu_1 & \mu_2 & \mu_3 \\ \mu_4 & 1+\mu_5 & \mu_6 \\ \mu_7 & \mu_8 & 1+\mu_9 \end{pmatrix}. \tag{B.1}$$

with $\boldsymbol{\mu} = [\mu_1 \quad \mu_2 \quad \mu_3 \quad \mu_4 \quad \mu_5 \quad \mu_6 \quad \mu_7 \quad \mu_8 \quad \mu_9]^T$ in $\mathcal{P}$, which is isomorphic to SL(3), the Special Linear Group of dimension 3 (Benhimane and Malis, 2007) satisfying $\det(\mathbf{H}) = 1$.

A group action $\mathbf{W}$, fulfilling equations (4), (5) and (6) can be defined from SL(3) to $\mathbb{P}^2$. For $\mathbf{H} \in \mathrm{SL}(3)$ and $\mathbf{x} = [x \quad y \quad 1]^T \in \mathbb{P}$, we have:

$$\begin{aligned} \mathbf{W} : \mathrm{SL}(3) \times \mathbb{P}^2 &\longmapsto \mathbb{P}^2 \\ (\mathbf{H}, \mathbf{x}) &\longmapsto \begin{pmatrix} \frac{\mu_1 x + \mu_2 y + \mu_3}{\mu_7 x + \mu_8 y + \mu_9} \\ \frac{\mu_4 x + \mu_5 y + \mu_6}{\mu_7 x + \mu_8 y + \mu_9} \\ 1 \end{pmatrix} \end{aligned} \tag{B.2}$$

Lie Algebra local reparameterization. For $\mathbf{H}$ around identity, $\mathbf{H}$ can be reparameterized using the matrix exponential map:

$$\mathbf{H}(\mathbf{v}) = \mathrm{expm}(\sum_{m=1}^{8} v_m \mathbf{G}_m) \tag{B.3}$$

with for any matrix $\mathbf{X}$:

$$\mathrm{expm}(\mathbf{X}) = \sum_{m=0}^{\infty} \frac{1}{m!}(\mathbf{X}^m) \tag{B.4}$$

and $\mathbf{v} = [v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6 \quad v_7 \quad v_8]^T$ in the Lie Algebra $\mathfrak{P}$ associated to $\mathcal{P}$. $\{\mathbf{G}_m\}_{m=1..8}$ are 3×3 matrix called Lie algebra generators. Under the condition $\det(\mathbf{H}) = 1$, the generators must satisfy:

$$\forall m, \quad \mathrm{Tr}(\mathbf{G}_m) = 0 \tag{B.5}$$

where Tr is the trace of a matrix. The $\mathbf{G}_m$ are the vector basis of the tangent space to the Lie Group manifold around the identity parameter $\boldsymbol{\mu} = \mathbf{0}$.

The $\mathbf{G}_m$ are chosen such that they convey a geometrical meaning. The generators associated to translations are :

$$\mathbf{G}_1 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{G}_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}. \tag{B.6}$$

Indeed, by applying (B.4) to $\mathbf{G}_1$ (resp. $\mathbf{G}_2$) we obtain the standard 3×3 matrix of a translation along the x-axis (resp. y-axis). The generators associated respectively to isotropic dilation about the origin and rotation about the origin are:

$$\mathbf{G}_3 = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & -1 \end{pmatrix} \qquad \mathbf{G}_4 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{B.7}$$

The generators associated to shear at 0 and 45 degrees are :

$$\mathbf{G}_5 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{G}_6 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{B.8}$$

The generators corresponding to the projective transformations induced by the parameters of the line at infinity are :

$$\mathbf{G}_7 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \qquad \mathbf{G}_8 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}. \tag{B.9}$$

Appendix C. Images Jacobian computation

In this section we tackle implementation issues about the Jacobian matrix associated to an image.

The Jacobian Matrix associated to I and T are

$$\mathbf{J}_{I,i}(\mathbf{0}) = \left. \frac{\partial I(\mathbf{W}(\boldsymbol{\mu}^k \circ \exp(\mathbf{v}), \mathbf{x}_i))}{\partial \mathbf{v}} \right|_{\mathbf{v}=\mathbf{0}} \tag{C.1}$$

and

$$\mathbf{J}_{T,i}(\mathbf{0}) = \left.\frac{\partial T(\mathbf{W}(\exp(\mathbf{v}), \mathbf{x}_i))}{\partial \mathbf{v}}\right|_{\mathbf{v}=\mathbf{0}}. \tag{C.2}$$

Using chain-rule, we can decompose these Jacobians as

$$\mathbf{J}_{I,i}(\mathbf{0}) = \mathcal{M}_I \mathcal{M}_{\mathbf{W}} \mathcal{M}_L \quad \text{and} \quad \mathbf{J}_{T,i}(\mathbf{0}) = \mathcal{M}_T \mathcal{M}_{\mathbf{W}} \mathcal{M}_L. \tag{C.3}$$

1. $\mathcal{M}_I$ (resp. $\mathcal{M}_T$) is a 1×3 matrix corresponding to the gradient of the warped image I (resp. T) with parameters $\boldsymbol{\mu}^k$ (resp. $\mathbf{0}$):

$$\mathcal{M}_I = \left.\frac{\partial I(\mathbf{W}(\boldsymbol{\mu}^k, \mathbf{x}))}{\partial \mathbf{x}}\right|_{\mathbf{x}_i} \quad \text{and} \quad \mathcal{M}_T = \left.\frac{\partial T(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}_i} \tag{C.4}$$

637 Those gradients can be implemented using the Söbel operator.

2. $\mathcal{M}_{\mathbf{W}}$ is a matrix corresponding to the Jacobian of the warp:

$$\mathcal{M}_{\mathbf{W}} = \left.\frac{\partial \mathbf{W}(\boldsymbol{\mu}, \mathbf{x}_i))}{\partial \boldsymbol{\mu}}\right|_{\mathbf{0}} \tag{C.5}$$

3. $\mathcal{M}_L$ corresponds to the Jacobian of the Lie algebra reparameterization:

$$\mathcal{M}_L = \left.\frac{\partial \exp(\mathbf{v})}{\partial \mathbf{v}}\right|_{\mathbf{0}} \tag{C.6}$$

638 $\mathcal{M}_{\mathbf{W}}$ and $\mathcal{M}_L$ can be precomputed before running the alignment algorithm.
639 $\mathcal{M}_T$ can also be precomputed. That is why the inverse approach using only
640 gradient of the template is computationally the most efficient of the ACL
641 approaches. $\mathcal{M}_I$ (and consequently $\mathbf{J}_{I,i}(\mathbf{0})$) has to be computed at each
642 iteration.

Particular case of homography. For the homography model

$$\mathcal{M}_{\mathbf{W}} = \begin{pmatrix} \mathbf{x}_i^T & \mathbf{0} & -x_i\mathbf{x}_i^T \\ \mathbf{0} & \mathbf{x}_i^T & -y_i\mathbf{x}_i^T \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \in \mathbb{R}_{3\times 9} \tag{C.7}$$

$$\mathcal{M}_L = [[\mathbf{G}_1]_v \quad ... \quad [\mathbf{G}_8]_v] \in \mathbb{R}_{9\times 8} \tag{C.8}$$

643 where $[\mathbf{G}_m]_v$ is the matrix $\mathbf{G}_m$ reshaped as a vector (in row major order) and
644 $\mathbf{x}_i$ are homogeneous image coordinates $\mathbf{x}_i = [x_i \quad y_i \quad 1]^T \in \mathbb{P}^2$.

References

Alter, F., Matsushita, Y., Tang, X., 2006. An intensity similarity measure in low-light conditions. In: European Conference on Computer Vision. Vol. 3954. pp. 267–280.

Authesserre, J.-B., Mégret, R., Berthoumieu, Y., April 2009. Asymmetric gradient-based image alignment. In: IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 981–984.

Baker, S., Matthews, I., 2004. Lucas-Kanade 20 years on: A unifying framework. International Journal of Computer Vision 56 (3), 221–255.

Bayro-Corrochano, E., Ortegón-Aguilar, J., 2007. Lie algebra approach for tracking and 3d motion estimation using monocular vision. Image and Vision Computing 25, 907–921.

Benhimane, S., Malis, E., July 2007. Homography-based 2d visual tracking and servoing. International Journal of Robotics Research 26 (7), 661–676.

Brooks, R., Arbel, T., 2009. Generalizing inverse compositional and ESM image alignment. International Journal of Computer Vision 87 (3), 191–212.

Dedeoglu, G., Kanade, T., Baker, S., May 2007. The asymmetry of image registration and its application to face tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (5), 607–623.

Dijk, J., van Eekeren, A. W. M., Schutte, K., de Lange, D. J., van Vliet, L. J., 2008. Superresolution reconstruction for moving point target detection. Optical Engineering 47 (9).

Guangwei, L., Yunpeng, L., Jian, Y., Zelin, S., 2008. Optimization on lie manifolds and projective tracking. In: IEEE International Conference on Computer Science and Software Engineering. Vol. 1. Los Alamitos, CA, USA, pp. 768–771.

Keller, Y., Averbuch, A., August 2004. Fast motion estimation using bi-directional gradient methods. IEEE Transactions on Image Processing 13 (8), 1042 – 1054.

Lucas, B., Kanade, T., 1981. An iterative image registration technique with an application to stereo vision. In: proc. International Joint Conferences on Artificial Intelligence. pp. 674–679.

Marey, M., Chaumette, F., May 2008. Analysis of classical and new visual servoing control laws. In: proc. IEEE International Conference on Robotics and Automation. Pasadena, California, pp. 3244–3249.

Mégret, R., Authesserre, J.-B., Berthoumieu, Y., 2008. The bi-directional framework for unifying parametric image alignment approaches. In: proc. European Conference on Computer Vision. pp. 400–411.

Miller, M. I., Younes, L., 2001. Group actions, homeomorphisms, and matching: A general framework. International Journal of Computer Vision 41 (2), 61 – 84.

Pham, T. Q., Bezuijen, M., van Vliet, L. J., Schutte, K., Luengo Hendriks, C. L., May 2005. Performance of optimal registration estimators. Vol. 5817 of proc. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference. pp. 133–144.

Shum, H.-Y., Szeliski, R., 2000. Construction of panoramic image mosaics with global and local alignment. International Journal of Computer Vision 36 (2), 101–130.

Vercauteren, T., Pennec, X., Perchant, A., Ayache, N., 2009. Diffeomorphic demons: Efficient non-parametric image registration. NeuroImage 45 (2), 561–572.