31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

Comparison of visual tracking algorithms on in vivo sequences for robot-assisted flexible endoscopic surgery

N. Masson, Fl. Nageotte, Ph. Zanne and M. de Mathelin
LSIIT UMR CNRS 7005 – Strasbourg University
{ masson, nageotte, zanne, demath}
@lsiit.u-strasbg.fr

J. Marescaux
IRCAD/EITS
jacques.marescaux
@ircad.u-strasbg.fr

Abstract— **Flexible endoscopes are used in many diagnostic and interventional procedures. Physiological motions may render the physicians task very difficult to perform. Assistance could be achieved by using motorized endoscopes and real-time visual tracking algorithm to automatically follow a selected target. In order to control the motors, one needs to have an accurate estimation of the motion of the target in the endoscopic view, which requires an efficient tracking algorithm. In this paper, we compare tracking algorithms on various in vivo targets in order to assess their behavior under different conditions. The study shows that among the difficulties which arise when tracking an in vivo target, the change of illumination is paramount. Nevertheless, some algorithms, with minor modifications and without a priori knowledge about the target, achieve very good results.**

I. Introduction

Medical procedures are nowadays less traumatic for the patient but often become more complex for the physicians. For instance, an increasing number of minimally invasive procedures use flexible endoscopes. The use of this type of device is not straightforward. For example, in angiomas burning procedures, gastro-enterologists encounter difficulties to follow the movement of the angiomas due to the patient's breathing and the stomach contractions. It may result in burning healthy tissues around the targeted angiomas. In newly developped surgical procedures, like Natural Orifice Transluminal Endoscopic Surgery (NOTES) [1], it is very difficult for the surgeon to operate the tip of the flexible endoscope using the handles and the flexible instrument in the working channel of the endoscope, in order to perform the surgical act and simultaneously track the physiological movements.

Motorized flexible endoscopes and gastroscopes can be a solution for improving existing and future procedures. In [2], the two wheels of the handle of an endoscope have been replaced by computer controlled motors. This system can be used to automatically track the target despite physiological motions and, thus, release the physician from this worry in order to focus on the procedure itself. The idea is that the computer identifies the displacement of a target chosen per operatively by the physician inside the image and controls the motors so that the tip of the endoscope follows this target. This way, the target would stay in the center of the endoscopic view.

In order to achieve this goal one has to be able to track in real-time the target in the image sequences. An additional constraint, because of the multiplicity of possible targets and the lack of a priori model, is the need to be able to define the target "in operam", without any marker.

There are many kinds of visual tracking algorithms in the literature [3]. One can separate them in two main categories depending on the type of features used to perform the tracking. The algorithms of the first kind use specific information of the target, like noticeable points or features in [4] and [5] or edges in [6]. They are referred as *feature-based* algorithms. The other kind of algorithms use all the pixels of the target and are referred as *area-based* algorithms. Such algorithms can be correlation based algorithms or histogram based algorithms. Area-based algorithms are more appropriate to our problem since they are not task specific and can be used by selecting any area of an image without a priori knowledge of the target. As stated above, with a great diversity of possible targets in surgical operations, it is, indeed, not relevant to have an algorithm trying to identify all the possible features of the targets. Besides, these features could also be modified during in vivo experiments. An interesting feature of this type of algorithms is the fact that they are generic and can be used for many different operations.

The aim of this paper is to compare real-time algorithms with different models and optimization techniques in order to study their behavior on in vivo sequences.

The first one is the Mean Shift algorithm which uses statistical analysis of the image as shown in [7]. All the others minimize a sum of squared differences (SSD) between two images, like Lucas-Kanade based algorithms [8] or the Efficient Minimization Method exposed in [9], which uses a new optimization method.

In the next section we will describe the in vivo sequences we have acquired to test the algorithms. These studied tracking algorithms are presented in section III. Section IV is dedicated to a comparison of the results on in vivo sequences. Finally, a discussion on these results will be done in the last section.

II. Description of the in vivo sequences

The four sequences of in vivo images, presented in Fig. 1, have been acquired in stomach and abdominal cavities of living pigs using a classical Karl Storz flexible endoscope. From these sequences five targets of possible interest for physicians have been identified. Two sequences have a

978-1-4244-3296-7/09/$25.00 ©2009 IEEE

periodical motion, we decided that three motion cycles were representative of the whole motion which makes the sequences 300 images long (12 seconds). For the two others we chose sequences of the same length with various motions.

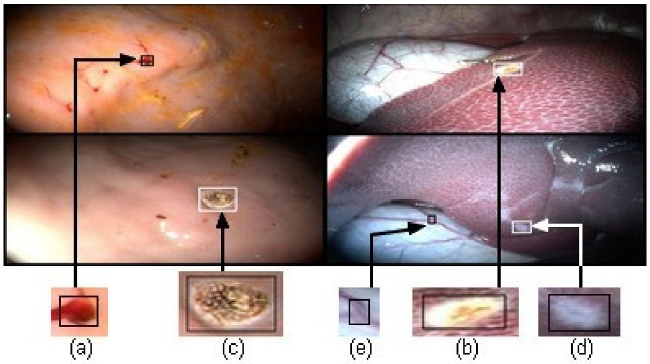

Fig. 1. The four in vivo sequences and the five targets used to test the algorithms. (a) target 1, (b) target 2, (c) target 3, (d) target 4, (e) target 5.

A. Target description

Let's describe the main transformations that occur to the five targets. In order to visualise more easily what happens in terms of pixels values, we have computed, for each target, the mean of the pixels values of the tracked Region of Interest (ROI), in every image. These values are shown on Fig. 2. Moreover, as the dominant tint on all sequences is red, we used greylevel images. Indeed in these conditions the additional information given by three channels images is not so relevant.

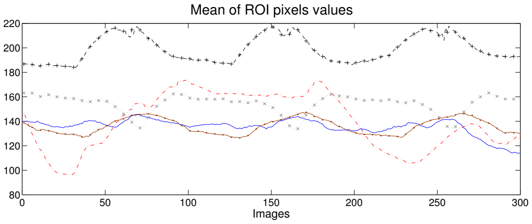

Fig. 2. Mean of the ROI pixels values undergone by the five targets used to test the algorithms. target 1: solid line. target 2: dashed line with $+$. target 3: broken line. target 4: solid line with $\bullet$. target 5: dotted line with $\times$.

The first target is a natural red blood spot on the pink pig stomach wall which becomes darker at the end of the sequence. The second target is a burnt on a pig liver which has a periodic beating motion. During its cyclic motion the target suffers a strong lighting as shows Fig. 2. The third target is a burnt in a pig stomach and has the most complex motion: first it moves to a darker area and comes back, then it undergoes a zoom out and a quick motion leading it again to the darker area. The fourth target is a pale natural mark on a pig liver and the fifth one is the base of a reddish vein fork on a mainly white pig gall bladder. Both have periodic motions but the last one suffers an important change of illumination.

B. Ground truth definition

To compare the algorithms accuracy we needed a ground truth on our in vivo sequences. For this purpose, on each of the targets in the 300 images long sequence, we have identified a specific point. Then we have manually selected the position of this point and recorded its coordinates in all the following images of the sequence. Hence we obtain a ground truth of the position of a specific point of the target that can be used for comparing the algorithms.

We choose for the template sizes $(width \times height)$, respectively from target 1 to target 5, 25×25, 60×30, 80×50, 30×20 and 15×15. These sizes enable to have the whole target inside the template. The size can be modified but if it gets too small the algorithm lacks information and if it gets too large the computation time increases. Overall the algorithm accuracy will fluctuate but it will not change dramatically.

III. Working of the tracking algorithms

In this section we shortly describe the algorithms. The first step, is the same for all these algorithms: the operator selects a rectangular area around the target to track in the image. This area is the reference template. Then the algorithms try to estimate the motion of this template in the subsequent images of the sequence. To assess their best possible behavior we let the algorithms converge on each image, whenever possible. Which means that, even if all the algorithms studied are known to work under real-time constraint, no computation time had to be respected during our experiments.

A. Mean Shift based tracking algorithm

The Mean Shift algorithm [7] is a pattern recognition method based on a probability density function (p.d.f.). The purpose is to find the mode of this function which corresponds to the highest probability of the new location of the target. The p.d.f. used is a greylevel histogram.

This algorithm is image based, which means it does not use the structural information of the image. Indeed only the pixels values are needed to compute the histograms which will be compared to the reference one. This algorithm is also known for being able to deal with deformations in the image.

From the selected template a reference histogram is computed. This histogram is the profile of our template. For each new image, a histogram is computed for an area centered at the position $\mathbf{y}_j$ (a 2D vector) the target had in the previous image. This area is a little wider than the reference template since, even if we consider the assumption of small displacement between two images, the target is supposed to have moved in an unknown direction.

The reference histogram to match is $\mathbf{q} = \{q_u\}_{u=1\dots m}$ and the current one is $\mathbf{p}(\mathbf{y}_j) = \{p_u(\mathbf{y}_j)\}_{u=1\dots m}$, with m the number of bins in these histograms. We have chosen $m = 32$ which allows a suitable repartition of pixels values from in vivo sequences.

Thanks to these histograms we compute a weight for each pixel $\mathbf{x}_i$

$$w_i = \sum_{u=1}^{m} \delta \left[b\left(\mathbf{x}_i\right) - u \right] v_u \sqrt{\frac{q_u}{p_u\left(\mathbf{y}_j\right)}} \qquad (1)$$

where the function $b : \mathbb{R}^2 \mapsto [1 \dots m]$ associates the bin $b\left(\mathbf{x}_i\right)$ to the pixel $\mathbf{x}_i$. v_u is a weighting value calculated from a background histogram, as in [10], to lower the effect of template pixels that are also present in the background: on the first image we create an histogram $\mathbf{h} = \{h_u\}_{u=1 \dots m}$ of pixels surrounding the template, then we calculate $\{v_u = min\left(h^*/h_u, 1\right)\}_{u=1 \dots m}$ where h^* is the smallest value of $\mathbf{h} = \{h_u\}_{u=1 \dots m}$. This suits particularly well our problem since the target we want to track is part of an organ and hence the background surrounding the target moves with it.

The mean shift equation is then written as

$$\mathbf{y}_{j+1} = \frac{\sum_{i=1}^{n} \mathbf{x}_i w_i}{\sum_{i=1}^{n} w_i} \qquad (2)$$

where $\mathbf{y}_{j+1}$ is the new location of the center of the template.

The new coordinates of the center is hence computed with the coordinates of all the pixels in the current template weighted with w_i from (1). These weights are such that the more influential coordinates come from the pixels whose brightness values are similar to the reference template ones (see [7] for details of the theory). The specificities of this algorithm compared to the other algorithms which will be detailled below is that it is not based on a transformation model between the ROI and the template, this also means that it cannot adapt its size if the target size is modified. In addition as the Mean Shift algorithm uses statistical analysis of the image pixels values it is known to be robust to distortion of the template.

B. Algorithms using a SSD

The algorithms based on a SSD use a minimization method to estimate the motion or transformation of a reference template $T(\mathbf{x})$ to a region of interest (ROI) being a part of the current image $I(\mathbf{x})$, where $\mathbf{x} = (x, y)^t$ is the coordinates vector of a pixel in the template. Let $W(\mathbf{x}; \mathbf{p})$ be a parameterized warp, where $\mathbf{p} = (p_1, \dots, p_n)^t$ is a vector of parameters. When applied to $I(\mathbf{x})$, W transforms it into $I'(\mathbf{x})$. To find $\mathbf{p}$, which maps $I(\mathbf{x})$ to $T(\mathbf{x})$, one has to minimize a non linear function over $\mathbf{p}$, which is the sum of squared errors between the current transformed template and the reference template:

$$\sum_{\mathbf{x}} \left[I\left(W(\mathbf{x}; \mathbf{p})\right) - T(\mathbf{x}) \right]^2 . \qquad (3)$$

1) Choice of the transformation model: Before solving (3) one needs to choose a transformation model which will define the warping function W. As the movements on our in vivo sequences may also undergo rotations and scale variations we have tested two models allowing such transformations. One is an affine warp written as in [11] :

$$W(\mathbf{x}; \mathbf{p}) = \begin{pmatrix} (1+p_1) & p_3 & p_5 \\ p_2 & (1+p_4) & p_6 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \qquad (4)$$

where $\mathbf{p} = (p_1, p_2, p_3, p_4, p_5, p_6)^t$ are the parameters to estimate. They allow to take into account a 2D translation, a 2D rotation, two scale factors and shear. The last model is a homography which allows to follow 3D motions of a planar patch:

$$W(\mathbf{x}; \mathbf{p}) = \begin{pmatrix} (1+p_1) & p_4 & p_7 \\ p_2 & (1+p_5) & p_8 \\ p_3 & p_6 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \qquad (5)$$

With the assumption for the template to be a planar patch the homography describes the 3D translation, the 3D rotation and the plane position with respect to the camera up to a scale factor.

2) Expression of the cost function: The Lucas-Kanade algorithm [12] solves (3) by iteratively estimating an increment $\Delta\mathbf{p}$ from a previously known estimate of $\mathbf{p}$ and then updating the parameters by $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ It is also called the forward additive algorithm [8]. Equation (3) has first to be linearized by a Taylor expansion leading to the local solution:

$$\Delta\mathbf{p} = H^{-1} \sum_{x} \left[\nabla I \frac{\partial W}{\partial \mathbf{p}} \right]^t \left[T(\mathbf{x}) - I\left(W(\mathbf{x}; \mathbf{p})\right) \right] \qquad (6)$$

where H is an approximation of Hessian matrix and $(\partial W / \partial \mathbf{p})$ is the Jacobian of the warp.

From then other cost functions have been developed. The forward compositional algorithm exposed in [13], aims at minimizing:

$$\sum_{x} \left[I\left(W\left(W(\mathbf{x}; \Delta\mathbf{p}) : \mathbf{p}\right)\right) - T(\mathbf{x}) \right]^2 \qquad (7)$$

and updating the parameters with $W\left(\mathbf{x}; \mathbf{p}\right) \leftarrow W\left(\mathbf{x}; \mathbf{p}\right) \circ W\left(\mathbf{x}; \Delta\mathbf{p}\right)$.

In [8] the authors implement another method, the inverse compositional which minimizes:

$$\sum_{x} \left[T\left(W\left(\mathbf{x}; \Delta\mathbf{p}\right)\right) - I\left(W\left(\mathbf{x}; \mathbf{p}\right)\right) \right]^2 \qquad (8)$$

with $W\left(\mathbf{x}; \mathbf{p}\right) \leftarrow W\left(\mathbf{x}; \mathbf{p}\right) \circ W\left(\mathbf{x}; \Delta\mathbf{p}\right)^{-1}$ as update for the parameters $\mathbf{p}$. This method is more efficient than the two above since most of the computations are done only once on $T\left(\mathbf{x}\right)$, the reference template, at the start of the algorithm.

3) Choice of the optimization method: At this point comes the question of the optimization method to use in order to solve (3). In [8], Baker and Matthews have shown that, among the well-known optimization techniques, the steepest descent and various diagonal approximations to the Hessian do not converge efficiently and are very sensitive to the parameterization for a same set of warps. The Newton algorithm is better but still worse than the Gauss-Newton or Levenberg-Marquardt algorithms. Since the results between the two latters are similar we have chosen to implement the

Gauss-Newton (G-N) optimization, which is more straightforward. This gives us the following approximation for the Hessian in (6), $H = \sum_x \left[\nabla I(\partial W/\partial \mathbf{p})\right]^t \left[\nabla I(\partial W/\partial \mathbf{p})\right]$.

In addition to the Gauss-Newton method we have tested another optimization method exposed in [9]. As classical second-order minimization perform worse than G-N method, E. Malis developed a different optimisation method called the efficient second-order minimization method (ESM). This method allows to approximate second-order terms by only calculating first-order terms. This makes the algorithm converge quickly and with little computation. Then (6) can be rewritten as :

$$\Delta \mathbf{p} \approx -2\left(J(\mathbf{0}) + J(\mathbf{p}_c)\right)^{+}\left[T(\mathbf{x}) - I\left(W(\mathbf{x};\mathbf{p}_c)\right)\right] \quad (9)$$

with $J(\mathbf{z}) = \partial I\left(W(x;\mathbf{p})\right)/\partial \mathbf{p}$ for $\mathbf{p} = \mathbf{z}$ and where $\mathbf{p}_c$ is the solution to equation (7).

4) Choice of parameterization: Besides classical parameterizations shown in III-B.1, there is a need for a specific one, indeed (9) generally requires to know $\mathbf{p}_c$. However, by parameterizing the homography matrix on a Lie algebra it is possible to compute (9) without explicitly knowing $\mathbf{p}_c$. Practically, $J(\mathbf{p}_c)$ is computed as the current image gradient $\nabla I(\mathbf{x})$ times the derivative of the warp with respect to the parameters $\partial W(\mathbf{x};\mathbf{p})/\partial \mathbf{p}$ computed for $\mathbf{p} = 0$. For further detailed explanations see [14]. Though this parameterization using Lie algebra could be applied to all transformation models of III-B.1, we only implemented it for the homography.

The Lie algebra parameterization allows to express the computation of the parameter increment using the mean of the gradients of the current image and the template, $\frac{1}{2}(\nabla I + \nabla T)(\partial W/\partial \mathbf{p})$. Although this is not theoretically exact we also tried to use the same computation of the mean of gradients for a classical affine parameterization.

5) SSD-based tested algorithms and notations: Many different combinations are possible between the various models, costs functions, optimization methods and parameterizations. We will now sum up which combinations have been chosen. There are the affine model with G-N and forward compositional (AFF+FC), the same with inverse compositional (AFF+IC), as well as the one with mean of gradients applied on forward compositional (AFF+MGF). The homography model with G-N and forward compositional (HMG+FC), also with inverse compositional (HMG+IC) or with parameterization on a Lie algebra (HMG+FCLie) and the optimization method developed by Malis with homography model (ESM).

IV. Experimental results

There are mainly three possible outcomes when testing an algorithm on a target: either it manages to track,with more or less accuracy, or it drifts, meaning it tracks the target for a while then looses it but keeps moving in the images with a smooth motion, or it fails: this happens when the ROI detected by the algorithm exits the image or when the ROI folds on itself and then does not move any more.

Each algorithm, has been initialized to the ground truth position on the first image of the sequence. The tracking accuracy is estimated by comparing the mean and standard deviation of the distance between the position estimated by the algorithm and the reference position from the ground truth. In the following tables the symbol ∅ indicates that the algorithm failed to track the target.

A. Comparing classical homography model algorithms

Tab. I shows the results for HMG+IC, HMG+FC and HMG+FCLie, Tab. II the results of ESM.

HMG+FC and HMG+FCLie have very close behavior. This is also true for HMG+IC and HMG+ICLie (not shown here). Both fail on the same targets. For the following of this section we will only consider the traditionally paremeterized version of the HMG algorithms.

HMG+IC and HMG+FC do not behave the same way. HMG+IC fails on much more targets than HGM+FC. The reason seems to be the changes in the ROI illumination, as seen on Fig. 2. When the target does not suffer an illumination change HMG+FC and HMG+IC track similarly. Tab. I also shows that, when a change of illumination occurs, even if HMG+IC does not fail, it does not track accurately. The difference between HMG+FC and HMG+IC is that the Jacobian matrix is computed at every iteration for HMG+FC whereas it is only computed on the reference template for HMG+IC. We will hence only consider HMG+FC in the following as it can be applied with better results on more targets.

When comparing the HMG+FC with the ESM we note that even with a different parameterization and a different optimization method, the results are very similar for targets 1, 3 (where they both fail), 4 and 5. On target 2 the ESM has a smaller error than the HMG+FC (or HMG+FCLie) which may indicate that the optimization of ESM works more efficiently when there is important lighting variations.

B. Comparing classical affine model algorithms

With affine model we observe, on Tab. II, similar differences between forward and inverse compositional algorithms. The AFF+IC fails to track some targets when there is an illumination change (usually a darkening). When there is an important increase of lighting (see target 2) AFF+IC does not fail but the tracking is badly impaired while AFF+FC is very close from the ground truth. So for the affine model, just as for the homography model, the forward compositional is more robust on in vivo sequences.

Concerning the new algorithm AFF+MGF, which is only slightly different from AFF+FC, we observe that on targets where AFF+FC tracks correctly it achieves similar results, see Tab. II. However AFF+MGF obtains much better results on target 3 where AFF+FC, the best algorithms on the sequence, drifts after the zoom out, see Fig. 3. Indeed using a mean of gradient seems to enable the optimization algorithm to step out of a local minimum allowing AFF+MGF to track the target until the sequence end. So AFF+MGF always manages to track the target it is assigned to, whatever the conditions and without estimation of the illumination.

Target \ Algorithm	Mean Shift	HMG+IC	HMG+FC	HMG+FCLie
Target 1	2.67 ± 0.85	∅	1.71 ± 0.71	1.71 ± 0.70
Target 2	3.22 ± 2.74	8.15 ± 9.91	2.57 ± 3.86	2.56 ± 3.85
Target 3	141.86 ± 49.92	∅	∅	∅
Target 4	4.63 ± 1.44	3.20 ± 0.67	3.18 ± 0.67	3.18 ± 0.67
Target 5	10.56 ± 4.46	∅	2.27 ± 1.01	2.26 ± 1.01

TABLE I

MEAN AND STANDARD DEVIATION OF DISTANCE TO THE GROUND TRUTH FOR MEAN SHIFT, HMG+IC, HMG+FC AND HMG+FCLIE.

Target \ Algorithm	AFF+MGF	AFF+IC	AFF+FC	ESM
Target 1	1.78 ± 0.67	1.79 ± 0.65	1.76 ± 0.67	1.70 ± 0.72
Target 2	2.10 ± 1.80	12.34 ± 16.99	1.69 ± 1.28	2.18 ± 2.31
Target 3	1.70 ± 1.46	∅	54.34 ± 58.67	∅
Target 4	3.10 ± 0.68	3.10 ± 0.68	3.09 ± 0.68	3.18 ± 0.66
Target 5	2.69 ± 1.23	∅	2.18 ± 1.07	2.81 ± 1.49

TABLE II

MEAN AND STANDARD DEVIATION OF DISTANCE TO THE GROUND TRUTH FORAFF+MGF, AFF+IC, AFF+FC AND ESM.

C. *Comparing homography and affine models algorithms*

Tab. I and Tab. II show that HMG+IC is more accurate than AFF+IC on target 2, where increase of lighting occurs. Nevertheless the affine model seems to be more robust since it does not fail on target 1 when HMG+IC does.

Regarding HMG+FC and AFF+FC results tend to indicate that HMG+FC is more accurate when tracking. Here too the affine model with AFF+FC seems to be more robust since, on target 3, it does not fail in the darker area while HMG+FC does, even if it drifts after the zoom out, explaining the large values in Tab. II.

About the two algorithms using a mean of gradients, ESM and AFF+MGF, they both track accuratly most targets. Once more the target 3 gives us more information about their respective behavior showing that the affine model tracks accuratly when the homography fails.

D. *Behavior of the Mean Shift algorithm*

The last algorithm is the Mean Shift algorithm. It follows quite accurately the ground truth in the sequences, though not as precisely as the previous algorithms. When the target almost does not evolve during the sequence, like target 1, the results of the Mean Shift are as good as the ones of the other algorithms.

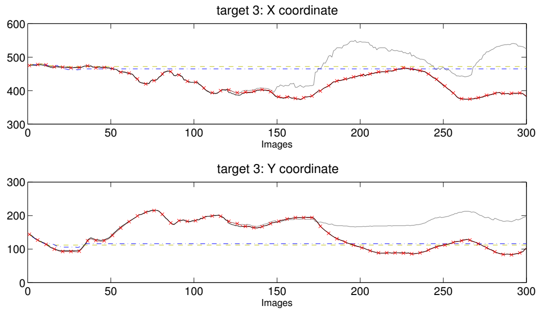

Fig. 3. Ground truth (solid black line) and results of tracking on target 3 for HMG+FC (dash and dot line), ESM (dashed line), AFF+FC (solid light line) and AFF+MGF (solid line with ×).

Furthermore, for different reasons, the Mean Shift algorithm has the same difficulties to track when illumination changes, see target 3 on Tab. I. Since only the values of pixels are used in this algorithm, it is quite sensitive to illumination changes. That is why it cannot follow the target 3 when it goes in the darker area. In addition, since the Mean Shift only uses the pixels values it is more sensitive to surrounding ones, all the more than our in vivo endoscopic images have low contrast. Hence the template size may influence the algorithm behavior. Indeed, even with the background rejection, when an area close to the target has a similar histogram (e.g. around target 4), the algorithm may drift if the template is too wide. The drift effect is even more readily understandable with target 5 where the Mean Shift algorithm moves the ROI along the vein. This indicates clearly that the Mean Shift algorithm can only be used for isolated targets, which is not so common in in vivo environment.

V. DISCUSSION

The mean shift algorithm achieve good results in tracking as long as there is no illumination change in the template and as long as the target does not change too much from its initial appearance. Otherwise, it becomes much less accurate or even drift far away from the target.

Among the SSD based algorithm which can estimate the target deformations, we found a few noticeable characteristics. Firstly, contrary to [8], we found out that the forward and inverse compositional algorithms are not equivalent on our in vivo sequences. This might be due to the fact that the target in the sequences often evolves, changing the illumination and preventing the inverse compositional algorithm to converge. Secondly, we found that the parametrization of the homography has little if any influence on the convergence of the forward compositional algorithm. Besides, this algorithm behaves in a very similar way to the ESM one which might suggest that such a complex optimization and parametrization in not needed on in vivo images.

On most of the targets there is not much difference between affine model and homography model. However, the affine model manages to handle strong darkening on target 3 when the other fails, regardless of the optimization method. This might suggest that the homography is over parameterized and, hence, could not manage to faithfully estimate all the parameters when less information is available from the gradients (e.g. because of darkening).

Finally, the affine model with mean of gradient (AFF+MGF) manages to almost always track quite accurately the target and presents the best results out of all the algorithms that have been tested.

To conclude, the illumination seems to have the most troublesome effect on the algorithms and one could think of estimating its parameters to improve their convergence. A method is explained in [15] about the ESM algorithm. In future work it would be worthy to apply such an estimation on the affine model with mean of gradient forward compositional algorithm to assess whether it improves its tracking ability. Another path which could also be used is to think of an appropriate way of updating the reference template. But a risk is to forget about the initial appearance of the target or to change too quickly on some temporary modifications of the target.

VI. ACKNOWLEDGMENTS

This work has been made possible thanks to a Ph.D. scholarship from Alsace Regional Council and to the company Karl Storz (Tuttlingen, Germany) which has provided the flexible endoscope.

REFERENCES

[1] S. Jagannath, S. Kantsevoy, C. Vaughn, and A. Kalloo, “Per-oral transgastric endoscopic ligation of fallopian tubes with long-term survival in a porcine model,” *Gastrointestinal Endoscopy*, vol. 61, no. 3, pp. 449–453, 2005.

[2] L. Ott, P. Zanne, F. Nageotte, M. de Mathelin, and J. Gangloff, “Physiological motion rejection in flexible endoscopy using visual servoing,” in *International Conference on Robotics and Automation*. IEEE, 2008.

[3] S. Hutchinson, G. Hager, and P. Corke, “A tutorial on visual servo control,” *Robotics and Automation, IEEE Transactions on*, vol. 12, no. 5, pp. 651–670, Oct 1996.

[4] T. T. H. Tran and E. Marchand, “Real-time keypoints matching: application to visual servoing,” in *Robotics and Automation, 2007 IEEE International Conference on*, April 2007, pp. 3787–3792.

[5] J. Shi and C. Tomasi, “Good features to track,” in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR ’94., 1994 IEEE Computer Society Conference on*, Jun 1994, pp. 593–600.

[6] P. Bouthemy, “A maximum likelihood framework for determining moving edges,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 5, pp. 499–511, May 1989.

[7] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” in *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR’00*, vol. 2, Hilton Head Island, USA, June 2000, pp. 142–149.

[8] S. Baker and I. Matthews, “Lucas-kanade 20 years on: A unifying framework,” *International Journal of Computer Vision*, vol. 56, no. 1, pp. 221 – 255, March 2004.

[9] S. Benhimane and E. Malis, “Real-time image-based tracking of planes using efficient second-order minimization,” in *IEEE Int. Conf. on Intelligent Robots and Systems, IROS’04*, vol. 1, Sendai, Japan, September 2004, pp. 943–948.

[10] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-based object tracking,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–575, 2003.

[11] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, “Hierarchical model-based motion estimation,” in *ECCV ’92: Proceedings of the Second European Conference on Computer Vision*. London, UK: Springer-Verlag, 1992, pp. 237–252.

[12] B. Lucas and T. Kanade, “An interactive image registration technique with an application in stereo vision,” in *In The 7th International Joint Conference on Artificial Intelligence*, 1981, pp. 674–679.

[13] H.-Y. Shum and R. Szeliski, “Construction of panoramic image mosaics with global and local alignment,” *Panoramic vision: sensors, theory, and applications*, pp. 227–268, 2001.

[14] E. Malis, “Improving vision-based control using efficient second-order minimization techniques,” in *ICRA*, 2004, pp. 1843–1848.

[15] G. Silveira and E. Malis, “Real-time visual tracking under arbitrary illumination changes,” in *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on*, Minneapolis, USA, June 2007, pp. 1–6.