Repetitive Motion Compensation for Real Time Intraoperative Video Processing

Michaël Sdika[a,*], Laure Alston[a], David Rousseau[a], Jacques Guyotat[b], Laurent Mahieu-Williame[a], Bruno Montcel[a]

[a] *Univ.Lyon, INSALyon, Université Claude Bernard Lyon 1, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1206, F69100, LYON, France*
[b] *Service de Neurochirurgie D; Hospices Civils de Lyon, Bron, France.*

Abstract

In this paper, we present a motion compensation algorithm dedicated to video processing during neurosurgery. After craniotomy, the brain surface undergoes a repetitive motion due to the cardiac pulsation. This motion as well as potential video camera motion prevent accurate video analysis. We propose a dedicated motion model where the brain deformation is described using a linear basis learned from few initial frames of the video. As opposed to other works using linear basis for the flow, the camera motion is explicitly accounted in the transformation model. Despite the nonlinear nature of our model, all the motion parameters are robustly estimated all at once, using only one singular value decomposition, making our procedure computationally efficient. A Lagrangian specification of the flow field ensures the stability of the method. Experiments on *in vivo* data are presented to evaluate the capacity of the method to cope with occlusion or camera motion. The method we propose satisfies the intraoperative constraints: it is robust to surgical tools occlusions, it works in real time, and it is able to handle large camera viewpoint changes.

Keywords: motion compensation, image registration, subspace learning, brain surgery, real time video processing, extended direct linear transform

[*] Corresponding author
Email address: Michael.Sdika@creatis.insa-lyon.fr (Michaël Sdika)

1. Introduction

Optical imaging is a modality of choice for surgery assistance: it is inexpensive, it has high temporal and spatial resolution and, as opposed to magnetic resonance or computational tomography, it implies only few constraints on the surgical material and room. This is why even simple devices such as true color camera are often used for assistance or monitoring during surgical intervention or radiotherapy for a variety of body parts: brain (Pichette et al., 2016), heart (Richa et al., 2011) or abdomen (Spinczyk et al., 2014) for example. Analysis of videos from these camera can be used for brain surface strain estimation (Ji et al., 2011), brain surface 3D reconstruction (Marreiros et al., 2016; Paul et al., 2009), to investigate the deformation of the exposed cortical surface (Ji et al., 2013), to measure hemodynamic response (Pichette et al., 2016; Villringer and Chance, 1997; Steimers et al., 2009; Sobottka et al., 2013; Oelschlgel et al., 2015). In all these works, motion compensation is used during some stage of the processing workflow with general purpose routines such as Horn and Schunck (Horn and Schunck, 1981) or Farnebäck (Farnebäck, 2003) optical flow routines, demons registration (Thirion, 1998) or the Matlab Medical Image Registration Toolbox. It is indeed mandatory to have very stable image in the surgical microscope when pixel-wise post-processing is needed. Even if the microscope is rather stable, this is not the case of the patient brain. Heart and respiratory pulsativities make the brain surface move. Furthermore awake surgery, used for example to investigate langages brain areas, implies strong unstability of the patient head and larger motion should be accounted for. The motion compensation can be improved by exploiting the specificity of the data and, as opposed to standard nonrigid registration methods, it should also be able to account for occlusions (of surgical tools for example). In this work, a motion compensation method is proposed that satisfy three important properties required to work in surgical conditions: the method is real time, it is robust to occlusions and the object motion is disentangled from the camera motion.

1.1. State of the Art

Literature on image registration is vast, allowing for instance multimodality registration with statistical cost function (Maes et al., 1997) or to guarantee that the deformation is invertible (Miller et al., 2002; Beg et al., 2005; Sdika, 2008, 2013). When it comes to motion estimation on video camera data, the high temporal resolution and the contrast invariance assumption simplify the problem: Methods in this case are mostly based on the optical flow equation (see (Fortun et al., 2015) a recent review).

In the problem of brain motion estimation during neurosurgery the same object is always present and undergoes a repetitive (but non periodic) non-rigid motion. By repetitive, we mean that the same deformation occurs repeatedly in the time, which is the case for the brain when the skull is opened. We do not make the stronger assumption of the motion periodicity. This allows accounting for temporal changes in the deformation caused for example by cardiac or respiratory rhythm changes. As the brain surface motion is similar all allong the video timecourse, it should be possible to find a representation of this motion with few degree of freedom. This led us to consider subspace learning to estimate a low dimensional space to describe this motion. Subspace learning has already been in several works (Black et al., 1997; Fleet et al., 2000; Irani, 2002; Roberts et al., 2009; Ricco and Tomasi, 2012; Wulff and Black, 2015; Sdika et al., 2016), to find a good representation of the deformable motion. It is assumed that the motion can be decomposed in a affine space. The basis of this space is found using a principal component analysis (PCA) on the motion of estimated on similar video. The dimension of the motion is reduced from twice the number of pixels to the number of principal components keeped (up to 500 in (Wulff and Black, 2015)). This allows to reconstruct the full motion from a reduced number of sparse keypoints To be robust to occlusion or tracking failures, M-estimation (Huber and Ronchetti, 1981; Ricco and Tomasi, 2012), expectation maximization (Roberts et al., 2009) or iterated reweighted least square (IRLS) (Wulff and Black, 2015) can be used.

In a previous work on the same application (Sdika et al., 2016), the motion is modeled using the sum of an affine transform (for the camera motion) and a deformable transformation (for the object deformation) which lies in a affine subspace estimated with a PCA. The camera image formation process implies however that the camera motion is a composition and not an addition to the deformable brain motion. The affine camera model used is also very restritive compared to the more realistic perspective model. In this work, solutions are proposed to these two problems.

1.2. Contributions

There are several contributions in this work. 1) We propose a motion model dedicated to the compensation of the repetitive brain motion during neurosurgery. The model explicitly distinguish between the camera motion and the nonrigid brain motion. We assume and experimentally check that the nonrigid brain motion lies in a low dimensional affine space that can be learned from few frames. 2) The main contribution is the resolution procedure. While the model is nonlinear, we will see that, with adequate variable changes, the equations involved to find our model parameters become considerably easier to solve: an original extension of the direct linear transform (DLT) (Abel-Aziz and Karara, 1971; Hartley, 1997) is proposed, involving only one singular value decomposition. These variable changes allow to avoid local minima considerations and to estimate the motion parameters easily. 3) From the robustness point of view, we propose a simple preprocessing of the keypoints using temporal consistency to detect outliers before parameters estimation. This enables to robustly estimate the parameters with only one fit. We also propose a procedure to correctly handle keypoints tracking under large camera viewpoint change.

Evaluation on real *in vivo* data has been done to investigate the influence of the parameters of our model and to compare our method to standard optical flow routines. The proposed procedure has also been evaluated into the framework of an *in vivo* protocol for intraoperative identification of somatosensory brain areas during neurosurgery.

2. Theory

2.1. Transformation Model

As far as we know, when a linear model is used for the optical flow as in (Black et al., 1997; Fleet et al., 2000; Ricco and Tomasi, 2012; Wulff and Black, 2015), no global motion component accounts for camera or global subject displacements. In this work, the camera motion is explicit. A Lagrangian representation is adopted and the geometric transformation between the initial and the t^{th} frame is modeled by

$$T(x,t) = U(t)T_d(x,t) \tag{1}$$

where $x \in \mathbb{R}^2$ is the spatial position of a given pixel in the initial frame, $t \in \mathbb{R}$ is the time and $T(x,t)$ gives the position of x in the t^{th} frame, $T_d(x,t)$ is the deformable part of the transform while the $U(t)$ transform accounts for the camera and global subject motion at time t. In this work, the pinhole camera model is used for the projection of the 3D scene on the camera and $U(t)$ is a homography (objects are approximated as planar surfaces). Using homogeneous coordinates (Szeliski, 2010) $T(x,t)$ and $T_d(x,t)$ are in $\mathbb{R}^3$, the homography $U(t)$ is a 3×3 matrix with 8 degrees of freedom and the homography applied to $T_d(x,t)$ is a simple matrix vector product.

It is assumed that the object deformable motion lies in a low dimensional affine space. This assumption will allow to use mostly linear expressions in our derivation and to have only few parameters to estimate. We consequently express the deformation T_d as a weighted sum of basis transformations:

$$T_d(x,t) = x + T_\mu(x) + \sum_{k=1}^{K} \lambda_k(t) p_k(x). \tag{2}$$

T_μ is the average local deformation, p_k are the stationary basis vectors of the deformation and $\lambda_k \in \mathbb{R}$ are the time dependent deformation coefficients. We want to stress again that despite the linear dependance of T_d on its parameters λ_k, it represents a nonlinear deformation of the image. The low dimensional affine subspace assumption has been made in several works on optical flow estimation. It is especially well adapted for the application we are interested in, because the object undergoes a repetitive deformation over time. It also allows to account for the camera motion separately from the brain surface motion and to learn a specific basis for each subject.

Once the basis vector p_k and T_μ are known, only $8+K$ degrees of freedom are left for the estimation of the transformation at a given timepoint: 8 for the homography and K for the λ_k coefficients. For our application, K is very low, around 5 in practice. The number of parameters to estimate is consequently reduced from twice the number of pixels for a standard optical flow routine to $8 + K$, less than 15.

Compared to other works using affine subspaces, the camera motion and the object deformable motion are explicit in our model. As the full transformation is the composition of these two terms, it depends on its parameters (U, λ_k) in a nonlinear fashion. At a first sight, this may seem problematic as it implies solving nonlinear optimization problems to estimate these parameters. We propose to use a first variable change to eliminate this problem. If $V(t) = U^{-1}(t)$, equation 1 becomes

$$V(t)T(x,t) = T_d(x,t). \tag{3}$$

We will see in the following sections that, consequently, all the steps of the registration process can be solved using linear algebra routines only.

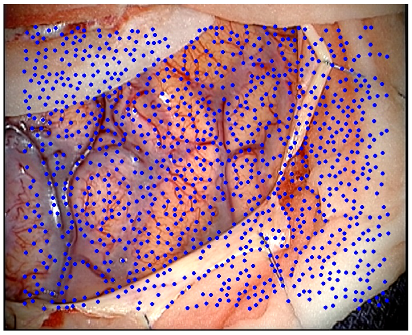


Figure 1: A frame with the Harris keypoints used for the tracking (distance between keypoints is $hd = 6$).

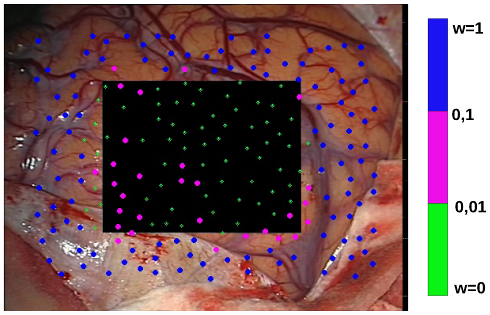


Figure 2: Keypoints (with $hd = 18$) colorcoded by their weight w in the presence of an occlusion.

2.2. Estimation of the Transformation

Assuming the deformation basis $(T_m u, p_k)$ is known, (details will be given in section 2.3), the transformation between the first frame and a new frame is estimated by fitting the parameters U and λ_k (or equivalently V and λ_k). In this section, the fitting procedure is described and as we will see, it is both efficient (only one SVD) and robust to occlusions.

2.2.1. Sparse Parameter Estimation with Extended DLT

As the degree of freedom for a transformation is now very low, it is not necessary to use all the pixels to estimate the parameters V and λ_k. Similarly to (Ricco and Tomasi, 2012; Wulff and Black, 2015), Harris keypoints (Harris and Stephens, 1988) are detected on the first frame and tracked along the frames with a sparse version of the Lucas and Kanade method (Bouguet, 2001). A more uniform distribution of the keypoint is obtained by setting the Harris threshold to 0 and by imposing a minimal distance between keypoints. A zero Harris threshold also makes the detection invariant to a global illumination change (see (Faille, 2003)). One can consider now that for each keypoint x_l detected on the first frame, its position at time t, $T(x_l, t)$, is known. In the figure 1, a frame is presented with the keypoint overlaid.

Using equation 3 on the keypoints at given time t, we obtain:

$$VT(x_l) - \sum_{k=1}^{K} \lambda_k p_k(x_l) = x + T_\mu(x_l), \tag{4}$$

where the dependency to t has been dropped for readability. From this set of equations (two for each keypoints), the transformation parameters V and λ_k can be estimated using the original extension of the DLT method proposed below.

Extended DLT estimation. The DLT (Abel-Aziz and Karara, 1971; Hartley, 1997) is a standard method to find an homography given a set of corresponding points in two images. To solve the equation 4, althought the correspondance are known, the DLT cannot be used out of the box to estimate V due to the unknown λ_k variables. The following extended DLT allows to estimate V and λ_k all at once. For each point x, let's define $T'_\mu(x) = x + T_\mu(x)$. Let's $c \in \{1, 2\}$ indexes the two scalar components of the 2D equation 4 and let's use superscript to denote the component of vectorial variables. Each scalar component of equation 4 can be written as:

$$\frac{v_{c1}T^1(x) + v_{c2}T^2(x) + v_{c3}}{v_{31}T^1(x) + v_{32}T^2(x) + v_{33}} \quad = \quad T'^c_\mu(x) + \sum_{k=1}^{K} \lambda_k p^c_k(x),$$

or equivalently:

$$\begin{aligned}
v_{c1}T^1(x) + v_{c2}T^2(x) + v_{c3} &= \big(v_{31}T^1(x) + v_{32}T^2(x) + v_{33}\big) \left(T'^c_\mu(x) + \sum_{k=1}^{K} \lambda_k p^c_k(x) \right) \\
v_{c1}T^1(x) + v_{c2}T^2(x) + v_{c3} &= v_{31}T^1(x)T'^c_\mu(x) + v_{32}T^2(x)T'^c_\mu(x) + v_{33}T'^c_\mu(x) \\
&+ \sum_{k=1}^{K} \big(\lambda_k v_{31}T^1(x)p^c_k(x) + \lambda_k v_{32}T^2(x)p^c_k(x) + \lambda_k v_{33}p^c_k(x)\big) .
\end{aligned}$$

For more condensed writing, let's define $T''^{ic}(x) = T^i(x)T'^c_\mu(x)$ and $p'^{ic}_k(x) = T^i(x)p^c_k(x)$. To eliminate the nonlinear interaction between the λ_k and the v_{3l} another variable change is done by defining: $\lambda'^l_k = \lambda_k v_{3l}$. The equations become linear:

$$\begin{aligned}
v_{c1}T^1(x) + v_{c2}T^2(x) + v_{c3} &= v_{31}T''^{1c}_\mu(x) + v_{32}T''^{2c}_\mu(x) + v_{33}T'^c_\mu(x) \\
&\quad + \sum_{k=1}^{K} \big(\lambda'^1_k p'^{1c}_k(x) + \lambda'^2_k p'^{2c}_k(x) + \lambda'^3_k p^c_k(x)\big)
\end{aligned}$$

Considering all the keypoints x and $c \in \{1, 2\}$, we end up with a linear system $Rv' = 0$ where the unknown vector $v' \in \mathbb{R}^{9+3K}$ is $v' = (v, \lambda') =$

$\left(v_{ij}, \lambda_k'^{l}\right)$ and $\lambda_k'^{l} = \lambda_k v_{3,l}$. Like v, v' is defined up to a scale factor. As with the standard DLT, for a given weight matrix W, v' is chosen as the solution of

$$\min_{\|v'\|=1} \|Rv\|_W ,$$

which is given by the eigenvectors corresponding to the smallest eigenvalue of $R^T W^2 R$. If $\bar{v}' = (\bar{v}, \bar{\lambda}')$ is one such eigenvector, the initial unknown parameters can be retrieved by solving the overdetermined system:

$$\begin{cases} v_{ij} &= \bar{v}_{ij} \\ \lambda_k v_{3l} &= \bar{\lambda}_k'^{l} \end{cases}$$

An analytical solution is available, if the first set of equation is solved exactly for v and the second set is solved in the least square sense for λ:

$$\begin{cases} v &= \bar{v} \\ \lambda_k^c &= \frac{\sum_l \bar{v}_{3,l} \bar{\lambda}_k'^{cl}}{\sum_l \bar{v}_{3,l}^2} . \end{cases}$$

2.2.2. Tracking keypoints under Large Camera Motion

The transformation model (equation 1) proposed in this work explicitly includes perspective transform and thus, it already allows large camera motions.

However, in a Lagrangian setting, the Lucas and Kanade algorithm can be problematic when the camera undergoes a large rotation: the windows used to solve the optical flow equations can be significantly rotated. While it is reasonable to assume a small change between successive frames, this assumption is false between the initial and the current frame. If a known homography allows to globally realign the two images, the following simple tracking procedure denoted as LRLK for Large Rotation Lucas and Kanade algorithm is applied: the second image is resampled and the starting value adjusted with the homography, the standard Lucas and Kanade algorithm is run with these new data and the resulting position is sent back with the homography. Thanks to its Lagrangian formulation, this procedure is stable: the errors do not accumulate from one frame to another; similarly to the Eulerian approach, it also allows large rotations and large camera motions in general. If for two images I_1 and I_2

$$q_2 = LK(I_1, q_1, I_2, q_2^0)$$

denote the position in the image I_2 of the points $q_1 \in I_1$ as given by the sparse Lucas and Kanade algorithm run with q_2^0 as starting value, the procedure is formally described in algorithm 1.

Algorithm 1 Large Rotation Lucas and Kanade's algorithm

```
procedure $LRLK(I_1, q_1, I_2, q_2^0, U)$
    $\bar{I}_2(x) = I_2(Ux)$
    $\bar{q}_2^0 \quad = U^{-1}(q_2^0)$
    $\bar{q}_2 \quad = LK(I_1, q_1, \bar{I}_2, \bar{q}_2^0)$
    $q_2 \quad = U\bar{q}_2$
    return $q_2$
```

As the global perspective transform between the initial and the current frame appears explicitly in our model (variable U in the equation 1), this procedure fits well within our framework: the homography estimated at the previous frame is used for the LRLK routine. The keypoints positions at time t, $q(t)$ are computed by

$$LRLK\left(I(.,0), q(0), I(.,t), q(t-1), U(t-1)\right).$$

2.2.3. Outliers Detection

If equation 4 is solved using a standard unweighted least square, the L_2 norm will make this estimation sensitive to outliers. Keypoints are outliers if their tracking fails for some reason, for example, due to the occlusion of the scene by an external object. In related works, the outliers problem is accounted using robust fit such as M-estimator (Ricco and Tomasi, 2012; Wulff and Black, 2015), or IRLS (Roberts et al., 2009; Sdika et al., 2016). In this work, we propose to use the temporal smoothness of the transformation to detect the outliers before the fit. Concretely, for a given keypoint x_l, equation 4 is weighted to remove the influence of points whose position changes abnormally fast. V and λ_k are estimated with a weighted least square fit where the x_l point equation for the fit at time t is weighted by:

$$w_l(t) = \exp\left(-\frac{\|T(x_l,t) - T(x_l,t-1)\|_2^2}{2\sigma_O^2}\right) \tag{5}$$

where σ_O is used to set the tolerance of what is an acceptable displacement between two frames. To summarize, when traking fails for some keypoints, they are given a lower weight in the fit. Motion parameters are estimated without these bad points while the global PCA model allow to recover the whole image deformation. Only one single weighted least square is necessary to robustly estimate the transformation parameters even with the presence of outliers. We will see however in section 3.4, that a few reweighting iterations have a low computational impact and can produces more visually pleasing results. In figure 2, a frame with occlusion is presented with the keypoints colorcoded by their weight. It can be clearly seen that the simple weighting scheme proposed allows to detect tracking failure due to occlusions.

2.2.4. Full pipeline summary

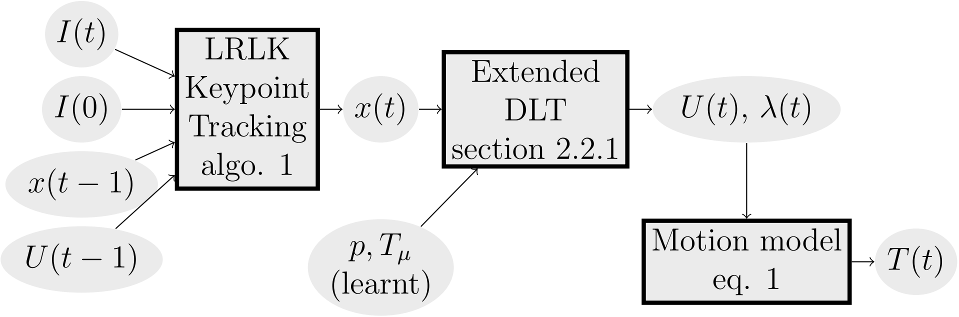


Figure 3: Pipeline of real time the motion estimation method after training. I: image frame, x: set of keypoints positions, U:camera motion, T_μ, p: motion basis learned on initial frames, λ: deformable brain motion parameters, T:dense motion.

To summarise, the brain motion compensation method proposed in this paper can be decomposed in two steps.

The first step is to learn the affine subspace in which the brain motion belong. To do so, the motion is estimated using a standard routine on few initial frames of the video. The raw brain motion is then extracted from each dense motion vector (section 2.3) and the principal modes of the brain motion are estimated with a PCA.

Once the brain motion basis is estimated, our fast method can be applied (see the pipeline on figure 3). Keypoints estimated of the first frame are tracked with the LRLK algorithm (algorithm 1). The extended DLT (section 2.2.1) is then used to recover the motion parameters. These parameters enables to compute the full dense motion (given by equation 1) in the given frame.

2.3. Learning the Nonrigid Transformation Basis

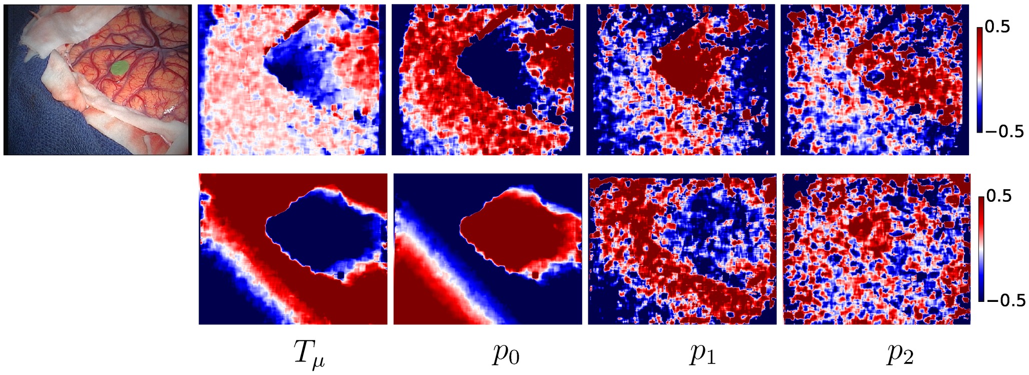


Figure 4: Images learned to model the brain deformable motion T_d (eqn. 2): mean displacement $T\mu$ and three first eigenimages p_k. First row is for displacement along x, second row is for displacement along y.

The first N_{tr} frames of the video are used to learn the affine subspace in which T_d lies, i.e. to estimate T_μ and p_k. First, $(T(.,t_i))_{i\in[1,N_{tr}]}$ are estimated with a standard motion estimation routine from the literature. Then, to extract the brain motion $T_d(.,t_i)$ from $T(.,t_i)$, the camera motion is estimated by solving

$$\min_{V(t_i)} \frac{1}{2} \sum_{x\in\mathcal{P}} \|V(t_i)T(x,t_i)\|_2^2 \,, \tag{6}$$

where $\mathcal{P}$ is a set of pixels and small deformations are assumed. Here again, Harris keypoints are used to reduce the computation time.

The problem 6 is a least square problem solved with standard geometric transform estimation routine for each timepoint.

Once the $V(t_i)$ are estimated, the deformable training transforms are given by

$$T_d(x,t_i) = V(t_i)T(x,t_i)$$

for $i \in [1, N_{tr}]$. These deformable transformations are assumed to lie in a low dimensional affine space. A basis of this subspace is estimated using a PCA to retain the space that best captures the variability of these vectors: T_μ is the mean of the $(T_d(.,t_i))_i$ vectors and p_k are the K first eigenvectors of their covariance matrix. An example of mean deformation and basis images is presented in figure 4.

3. Experiments and Results

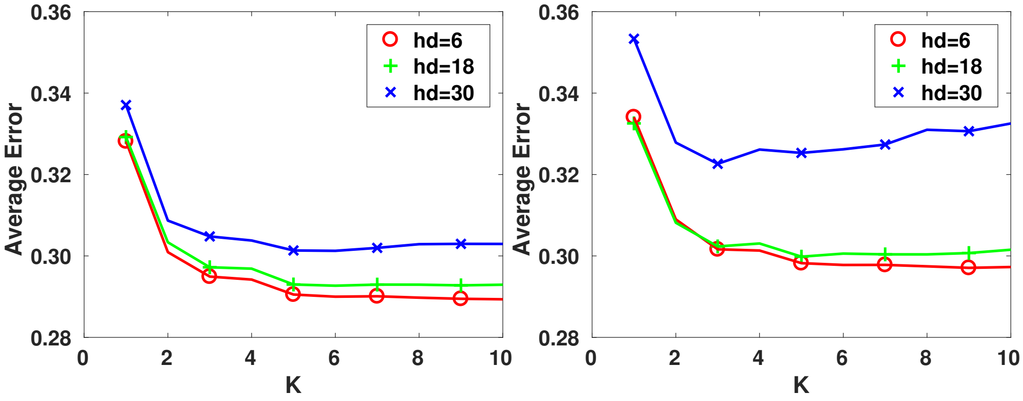
In this section, the proposed method in this paper has been evaluated on real neurosurgical videos taken in the operating suite. Experiments were done in the neurosurgery service of the Hospices Civils de Lyon. Twelve videos from three patients have been used in the experiments. The participating patients signed written consent. The study respects the Hospices Civils de Lyon local research ethics committee rules. Inclusion criteria were preoperative diagnosis of low grade glioma or high grade glioma; tumor judged suitable for open cranial resection; age equal to or older than 18 years; and patient ability to provide written consent. A 3-chip RGB CCD camera mounted on a clinical surgical microsocope (Zeiss OPMI Pentero) was used to record the video of opened brain areas. The microsocope objective was positionned around 25 cm above the brain. The image plane was positionned to correspond to the plane tangent to the center of the brain area exposed. The brain was illuminated by a xenon lamp with a direction making a small angle with the optical axis to optimize the light collection and specular reflexion attenuation compromise. Each video lasts between 30 and 90 seconds, has a frame rate of 25 fps and a frame size of either 511x388 or 720x576.

The Farnebäck method (Farnebäck, 2003), denoted as GF, and the total variation Perez method (Pérez et al., 2013), denoted as TV, as implemented in OpenCV 3.0[1], were used for comparison. The GF method was also used as the motion estimation routine on the training frames for the subspace estimation. The GF method has been chosen for its availability in CPU version in OpenCV and its relative efficiency. Note that other methods could have been chosen for comparison as long as they are also used for the training. For all but the training size investigation experiment, the 25 initial frames of each video were used as training set.

In section 3.1, the low dimensional affine model has been validated and the influence of its most important parameters has been investigated. In section 3.2, the robustness to occlusions of our method is evaluated. The robustness to large camera motions is finally evaluated in section 3.3. CPU considerations are presented in section 3.4. Finally, the use of our method into the framework of an *in vivo* protocol for intraoperative identification of somato-sensory brain area is illustrated in section 3.5.

[1] http://opencv.org

3.1. Parameter Analysis



(a) Weighted

(b) Unweighted

Figure 5: Mean error versus number of PCA component for several minimal distance between the keypoints (hd). Results are given when outliers weighting (eqn. 5) is used (Weighted) or not (Unweighted).

As no ground truth is avalaible, a really satisfying and absolute error metric cannot be used. As often, for this kind of work, we have to rely on derivative error measure. Here, the error is measured by

$$Err(T) = \frac{1}{n|\mathcal{B}|} \sum_{t=1}^{n} \sum_{x \in \mathcal{B}} \left\| T(x,t) - T_{\mathrm{GF}}(x,t) \right\| ,$$

where n is the number of frames in the video and $\mathcal{B}$ is a brain mask manually segmented on the initial frame and T_{GF} is the transformation obtained using the GF method. In the absence of ground truth, this measure is an indication of the adequacy of the model to the problem and that transformations obtained with high degrees of freedom methods can be represented by our low dimensional model.

In figure 5, the average error is plotted against the number of principal components for several values of the minimal distance (hd) allowed between the Harris keypoints, with and without weighted fit (eqn. 5). As expected, the method gives better results when hd is the lowest, or equivalently, when the number of keypoints is the highest. One can also see that, regarding K, the number of principal components, a plateau is quickly attained: the deformable component of the transformation indeed lies in a low dimensional

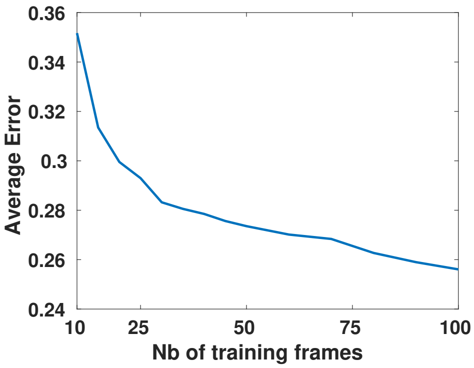


Figure 6: Mean error as a function of the number of training frames (using $K = 5$, $hd = 6$).

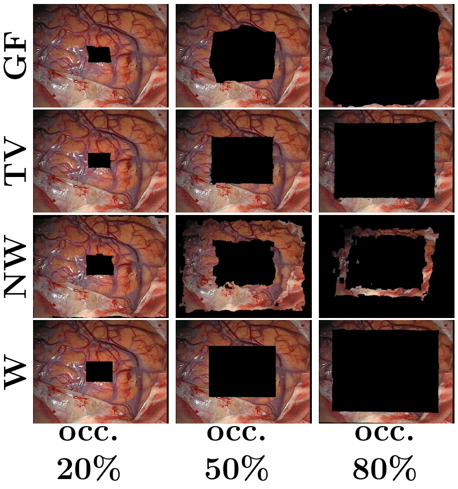


Figure 7: Snapshot of the motion compensated video for several percentage of occlusion. Results are presented for the opencv GF and TV methods and our method when outliers weighting (eqn. 5) is used (W) or not (NW).

affine space. However, when hd is too high, the method seems less stable and the error tends to increase again with K. Although there were no occlusions on the video we used, the weighting (eqn. 5) of the keypoints does improve the estimation considerably. The error globally decreases and stability is preserved with less keypoints.

The mean error as a function of the number N_{tr} of training frames is presented in figure 6 with $K = 5$ and $hd = 6$. The error quickly decreases for low N_{tr}, the decrease is less important when N_{tr} is greater than 40. Setting this parameter is a compromise between the desired accuracy and the training time: while a higher N_{tr} improves the accuracy, it also implies more use of the slower standard optical flow routine and that the high dimensional PCA will be done on a larger set.

For reference, the average difference between GF and TV is 0.27 ± 0.10. There is *a priori* no reason why GF or TV would be better but these results show that the difference between the proposed method and GF seems in the range of differences between established optical flow methods.

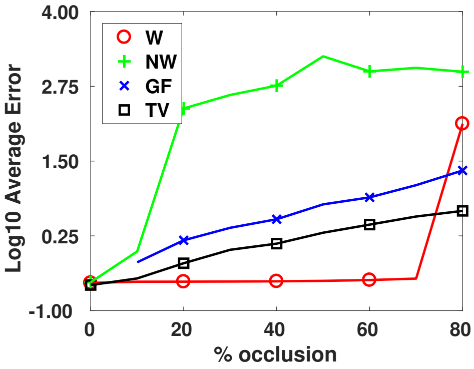


Figure 8: log_{10} mean error versus percentage of brain occlusion. Results are given for the opencv GF and TV methods and for our method when outliers weighting (eqn. 5) with $hd = 6$, $K = 5$. is used (W) or not (NW).

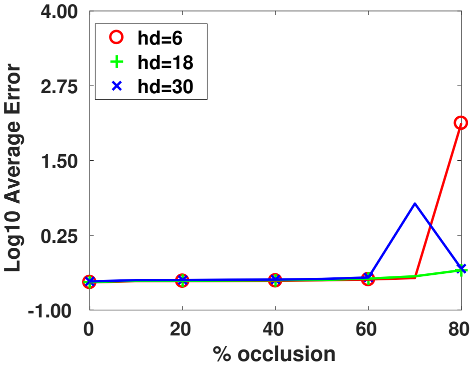


Figure 9: log_{10} mean error versus percentage of brain occlusion for several minimal distances between the keypoints (hd). Results are given when outliers weighting (eqn. 5) is used.

3.2. Video with Occlusion

In this section[2], a black rectangle has been placed in the center of the brain after 3 seconds until the end of the video to mimic the apparition of an object in the field of view of the camera. The experiment has been reproduced with a rectangle size varying from 10% to 80% of the brain. The error between the GF transform computed with no occlusion and the result of the motion estimation algorithms with occlusions is measured and presented in figures 8 and 9.

Snapshots of a motion compensated video are displayed in figure 7. Our method with the weighted fit produces non artifacted videos while the unweighted estimation implies strong artifacts all over the image. While maybe less visible on the snapshots, the video processed with the GF method presents smooth distortion artifacts in the vicinity of the rectangle. The video compensated using TV also presents artifacts in a smaller region and more sharply defined.

In the previous section, we saw that weighting the fit with equation 5 enables to lower the error and to improve stability with larger inter keypoints distance. On figure 8, one can see that this weighting is necessary when foreign objects enter the field of view. The fit fails even when the size of these

[2] A video showing results under occlusion is given in additional materials.

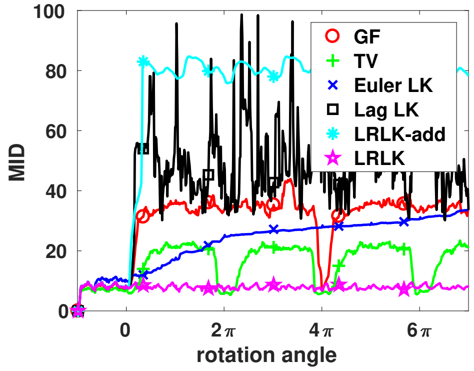


Figure 10: MID when the original video is rotated around the camera axis. Results are presented for the opencv Farnebäck routine (GF) and for our method when the tracking is done from one frame to the next (Euler LK), from the initial to the current (Lag LK), using our previous additive model (LRLK-add) and the proposed LRLK routine.

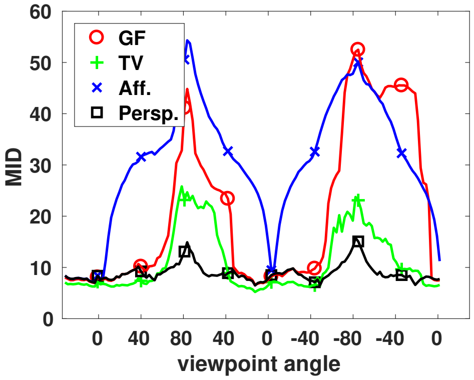


Figure 11: Average difference between the current and the initial frame when a time dependent viewpoint change is added to the video (in abscissa: the viewpoint angle). Results are presented for the opencv GF and TV routine and for our method with an orthographic (Aff.) or perspective (Persp.) camera model.

objects is small with respect to the brain size due to the high instability of the L2 norm to outliers. GF also fails in the presence of occlusions but in a lower extent. While not as robust as our method, TV is clearly more robust than GF. As expected, the results presented in figure 9 confirm that stability is better with low hd. It is however reassuring to see that even for quite large hd (30 pixels), the motion compensation still resists to about 60% or 70% of occlusions. In these cases, even if the motion compensation fails, in our experience, the fit get backs to normal when the object disappears.

3.3. Large Camera Motion

To assess both the validity of our model (eqn. 1) and the LRLK tracking (algorithm 1), we simulated camera motions by adding rotation, scaling or viewpoint changes to the input video after few seconds. We stress that the initial nonrigid motion is still present in the video with additional motion. The mean intensity difference (MID) between the current and the initial frame in the brain mask

$$MID(t) = \frac{1}{|\mathcal{B}|}\sum_{x\in\mathcal{B}} \|I(x,t) - I(x,0)\|_{\text{RGB}}\,,$$

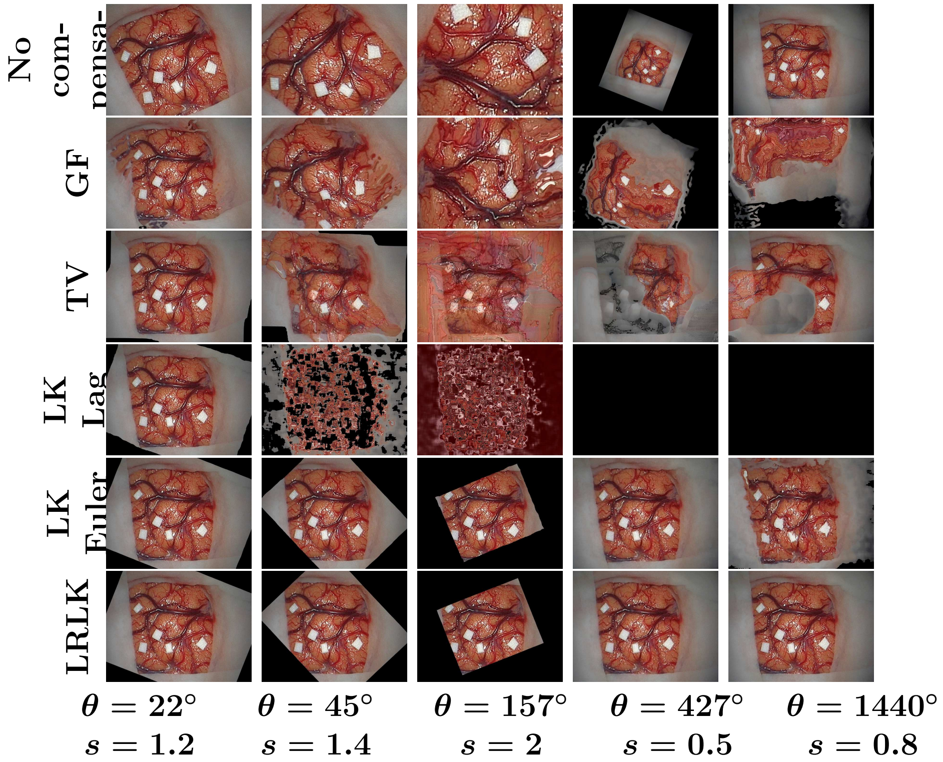


Figure 12: Snapshot of the motion compensation results when additional rotation/scale are added to the original video. Results are presented for the opencv Farnebaäck routine (GF) and for our method when the tracking is done from one frame to the next (LK Euler), from the initial to the current (LK Lag) and using the proposed LRLK routine.

where $\|.\|_{\text{RGB}}$ is the Euclidian norm in the RGB space, is used to compare GF, TV and variants of our method.

Pure rotation experiment results are presented in figure 10. A time dependent rotation is added to the initial video, the MID after compensation is plotted for GF, TV and our method with variants of Lucas and Kanade keypoint tracking: LK Lag is the tracking with a Lagrangian setting $q(t) = LK(I(0), q(0), I(t), q(t-1))$, LK Euler for an Eulerian specification $q(t) = LK(I(t-1), q(t-1), I(t), q(t-1))$ and LRLK is the algorithm 1 keypoint tracking $q(t) = LRLK\,(I(0), q(0), I(t), q(t-1), U(t-1))$. To assess the benefit of using the new physically meaningful composition model, we also included our previous additive model (Sdika et al., 2016) in the comparison: the camera motion is an affine transform that is added to the deformable brain motion. This model, denoted as LRLK-add is used with the LRLK tracking.

One can see on figure 10 that both GF and LK Lag rapidly explode. While TV is better than GF, it does not resist to even slight rotations. One can however notice that TV goes back to normal at every full rotation. In ”Lucas and Kanade like” methods, even if the transformation is correctly initialized from the previous frame, the local window around each pixel does not rotate and ends up being really different. The known drawback of the Eulerian estimation is visible: the LK Euler error does not explode but is slowly but surely increasing. With LRLK, the error is stable and stay at the level before the rotation begins. This performance is the result of the combination of two ingredients: the Lagrangian setting which provides the stability and the camera motion estimation and resampling which allows the large global perspective changes. One can also see that using the physically meaningful composition model is essential: despite the tracking is done with LRLK, the additive model LRLK-add fails for very small affine camera motion.

In figure 12 are displayed snapshots of motion compensation results when not only rotation but also scale changes (from 0.5 to 2) have been added[3]. Results of the previous paragraph are visually confirmed: neither GF, TV nor LK Lag are able to estimate the rotational motion of the camera; LK Euler is far better at first sight but on long time course, errors accumulate;when the tracking is done with LRLK, the motion estimation stays correct and the video is not artifacted even for a full rotation and scale changes.

For the next experiment, we simulated a viewpoint change in which the camera is moved around the pulsating brain. The MID is plotted in figure 11 as a function of the viewpoint angle. Our model with an orthographic camera (Aff.) and with a pinhole model (Persp.) is compared to TV and GF. For Aff, the only difference in our framework is to replace the homography U by an affine transform. Note that in this case, the extended DLT is not necessary: the equation 4 is linear and can be solved with linear least square routine. Both TV and GF are good up to ±45°. TV resists better than GF for high viewpoint changes but still, the results are not acceptable. One can see that the use of a perspective camera in our model is very important here: while the error is very high from the first viewpoint change degree with the orthographic camera, the pinhole camera model is very robust up to very large viewpoint changes.

[3] The video and the results are given in additional materials.

One can also remark that real life camera motions will include pure viewpoint change mixed with rotation and scaling. Errors from the different camera motion experiments will be added and the disprepancy between our method with a perspective model and the other methods should be even clearer.

3.4. CPU Time

	Single Core				Multi Core	
hd	GF	TV	$N_{IRLS} = 1$	$N_{IRLS} = 4$	$N_{IRLS} = 1$	$N_{IRLS} = 4$
6	6.99	0.12	23.81	21.74	56.13	49.28
18			26.67	25.97	59.15	57.40
30			28.17	27.40	63.27	61.34

Table 1: Number of frame per seconds for the OpenCV GF and TV methods, and our method with different parameters: N_{IRLS} is the number of least square re-weighting (all other experiments have been done with $N_{IRLS} = 1$), hd is the minimal distance between the pixels.

CPU measurements have been done on a Dell desktop with Intel Xeon E5-2640 2.40GHz processors. Measurements have been done using a single or 4 cores. The number of frames processed per second (fps) has been reported in table 1 for several variants of our method and the GF and TV method. Frame rate measurements includes reading from the input file, motion compensation and writing results to disk. Measurements have been done on 720 × 576 images. Frame rate of the input video was 25 fps.

Depending on the parameters chosen for our method, it is around 3 to 4 time faster than the GF method and these parameters can be chosen to have a real time motion compensation. The TV method is really expensive. The CPU criterion clearly discards this method for use in real clinical condition, even as the routine used for the training frames.

Among the two parameters hd and N_{IRLS} of our method, only the minimal distance between keypoints hd has a real influence on the computation time. This parameter is directly linked to the number of keypoints used in the processing. Although the processing involved with N_{IRLS} also depends on the number of keypoints, its influence on computation time is low because it does not depend on the LK tracking which represents a large part of processing time. The PCA to compute the T_d basis takes about 2 GF iterations.

Although the efficiency of our multi core implementation is only around 40%, it is sufficient to be clearly above the real time frame rate of 25 fps with only 4 processors and to be able to add further processing for the analysis of the video for clinical parameter extraction.

3.5. Application to intraoperative brain areas identification

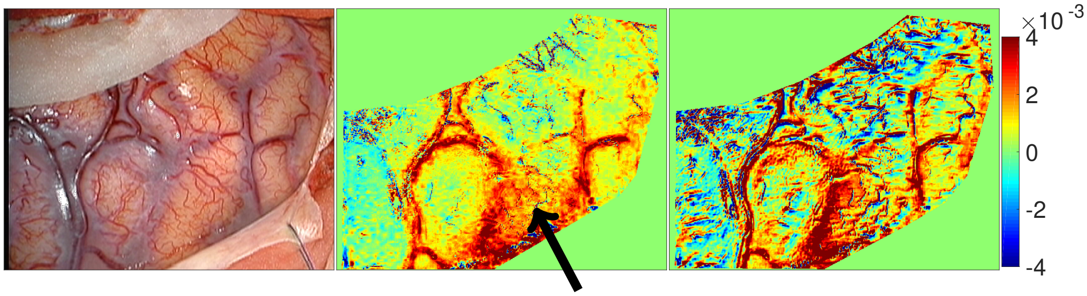


Figure 13: The screenshot of a video (left) and the corresponding hemodynamic maps computed with (middle) and without (right) our motion compensation method as pre processing. The map without motion compensation presents motion artefact near the blood vessels. Colorbar show variations in total hemoglobin concentration in arbitrary units. The black arrow reveals the somato-sensory cortex identified by cortical stimulation during surgery; it corresponds to area of high value in our maps.

As an illustration of the relevance of the proposed motion compensation method in a real clinical application, the intraoperative identification of somato-sensory brain areas during tumor resection in neurosurgery. These area are so far often identified using the potentially risky (epilepsy) electrostimulation method: brain surface areas are stimulated, if the stimulation of a given area triggers a finger motion, then the ”finger motion area” is identified. This identification method is used as gold standard and compared to a purely optical method: finger motions are applied to the anesthetized patient with a 30s rest; 30s stimulation; 30s rest paradigm. Variations in the hemodynamic parameters were then assessed based on the model described in (Jakovels and Spigulis, 2012), for blood variation assessment. This allows to compute a hemodynamic map to identify the somato-sensory cortex area corresponding to the finger motion.

Figure 13 shows the variations in total hemoglogin concentration between the first rest period and the stimulation period for each pixel. The activated area has been identified by the gold standard in clinical practice, that is electrical stimulation. Fig 13 illustrates that our motion compensation method is effective to dramatically decrease artifacts appearing all over the image and showing artificial hemoglobin variations. In particular the motion compensated image shows mainly green/yellow (no/slight variations) or red (hemoglobin increase due to brain activation) events. It should be opposed to the not compensated image which exhibits very noisy red and blue events. Therefore, while a red area corresponding to gold standard (electrical simulation) appears when the motion compensation is used, no clear activated brain areas can be identified without the motion compensation.

4. Conclusion

In this paper, we proposed a real-time motion estimation method to compensate repetitive brain motion on videos acquired during neurosurgery. Our method uses an original low degree of freedom transformation model defined as the composition of a perspective transformation and a nonrigid deformation lying on a low dimensional affine space. Despite the nonlinear nature of the transformation, two variable changes enables to transform the motion estimation into a much simpler problem solved with only a single SVD computation. The motion is estimated with the tracking of only a sparse set of keypoints for efficiency and using a Lagrangian specification for stability. Using the perspective part of our transformation model, we also proposed an adaptation of the Lucas and Kanade procedure able to reliably track the keypoints under large rotation of the camera around its axes and more generally, under large viewpoint change. Outliers are handled by weighting the least square: a single fit is sufficient to handle occlusions caused by foreign objects entering the field of view. Few reweighting iteration of the least square fit has however very little incidence in the computation time. Although the experiments in the paper have been done without these iterations, we found that in some extreme cases, adding these few iterations can help to produce visually better results at negligible cost.

As a clinical application of the proposed method we show its relevance as a preprocessing step in the intraoperative identification of brain areas during tumor resection. The usefulness of the proposed method is very clear to discriminate between real physiological hemodynamic events from noisy camera motion driven events. This step is then mandatory to realize a robust identification of brain areas in this demanding clinical context. The model used for areas identification was taken from (Jakovels and Spigulis, 2012).

Further works will consist in the improvement of this conversion model to get hemodynamic parameters from colorimetric variations.

In this article, the videos considered for illustration were acquired with an incoherent white light to produce RGB reflectance images of the human brain. Other optical setup using coherent light from lasers have also been reported to be useful during neurosurgery. This includes Laser Doppler imaging and Laser speckle imaging which can produce perfusion maps (Raabe et al., 2009; Richards et al., 2012; Parthasarathy et al., 2010). With such optical imaging modalities the blood vessels are also visible and the brain surface is highly textured. Therefore, the keypoints detection and tracking is possible and the real time motion compensation algorithm presented here could *a priori* also be applied with success in this extended field of brain imaging for neurosurgery.

The method has been evaluated on video recorded in neurosurgical unit. The motion estimation routine proposed in this paper can be done in real time, robustly and accounting for large viewpoint changes. This opens the way to further temporal analysis of these videos to help neurosurgeon in their action.

Acknowledgment

This work was supported by the LABEX PRIMES (ANR-11-LABX-0063) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).

References

Abel-Aziz, Y., Karara, H., 1971. Direct linear transformation from comparator coordinates into object space coordinates. Urbana, IL: American Society of Photogrammetry , 1–18.

Beg, M.F., Miller, M.I., Trouvé, A., Younes, L., 2005. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. International journal of computer vision 61, 139–157.

Black, M.J., Yacoob, Y., Jepson, A.D., Fleet, D.J., 1997. Learning parameterized models of image motion, in: Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on, IEEE. pp. 561–567.

Bouguet, J., 2001. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. Intel Corporation 5, 1–10.

Faille, F., 2003. Adapting interest point detection to illumination conditions., in: DICTA, pp. 499–508.

Farnebäck, G., 2003. Two-frame motion estimation based on polynomial expansion, in: Image Analysis. volume 2749 of *Lect. Notes in Comp. Sci.*, pp. 363–370.

Fleet, D.J., Black, M.J., Yacoob, Y., Jepson, A.D., 2000. Design and use of linear models for image motion analysis. International Journal of Computer Vision 36, 171–193.

Fortun, D., Bouthemy, P., Kervrann, C., 2015. Optical flow modeling and computation: A survey. Computer Vision and Image Understanding 134, 1 – 21. doi:http://dx.doi.org/10.1016/j.cviu.2015.02.008.

Harris, C., Stephens, M., 1988. A combined corner and edge detector., in: Alvey vision conf., p. 50.

Hartley, R.I., 1997. In defense of the eight-point algorithm. IEEE Transactions on pattern analysis and machine intelligence 19, 580–593.

Horn, B.K., Schunck, B.G., 1981. Determining optical flow, in: 1981 Technical symposium east, International Society for Optics and Photonics. pp. 319–331.

Huber, P.J., Ronchetti, E.M., 1981. Robust statistics. Wiley Series in probability and Statistics.

Irani, M., 2002. Multi-frame correspondence estimation using subspace constraints. International Journal of Computer Vision 48, 173–194.

Jakovels, D., Spigulis, J., 2012. Rgb imaging device for mapping and monitoring of hemoglobin distributionin skin. Lithuanian Journal of Physics 52.

Ji, S., Fan, X., Roberts, D., Paulsen, K., 2011. Cortical surface strain estimation using stereovision. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011 , 412–419.

Ji, S., Fan, X., Roberts, D.W., Hartov, A., Paulsen, K.D., 2013. Tracking cortical surface deformation using stereovision, in: Mechanics of Biological Systems and Materials, Volume 5. Springer, pp. 169–176.

Maes, F., Collignon, A., Vandermeulen, D., Marchal, G., Suetens, P., 1997. Multimodality image registration by maximization of mutual information. Medical Imaging, IEEE Transactions on 16, 187–198.

Marreiros, F.M., Rossitti, S., Karlsson, P.M., Wang, C., Gustafsson, T., Carleberg, P., Smedby, ., 2016. Superficial vessel reconstruction with a multiview camera system. Journal of Medical Imaging 3, 015001–015001.

Miller, M.I., Trouvé, A., Younes, L., 2002. On the metrics and euler-lagrange equations of computational anatomy. Annual review of biomedical engineering 4, 375–405.

Oelschlgel, M., Meyer, T., Sobottka, S.B., Kirsch, M., Schackert, G., Morgenstern, U., 2015. Intraoperative identification of somato-sensory brain areas using optical imaging and standard RGB camera equipment a feasibility study. Current Directions in Biomedical Engineering 1. doi:10.1515/cdbme-2015-0066.

Parthasarathy, A.B., Weber, E.L., Richards, L.M., Fox, D.J., Dunn, A.K., 2010. Laser speckle contrast imaging of cerebral blood flow in humans during neurosurgery: a pilot clinical study. Journal of biomedical optics 15, 066030–066030.

Paul, P., Morandi, X., Jannin, P., 2009. A surface registration method for quantification of intraoperative brain deformations in image-guided neurosurgery. IEEE Transactions on Information Technology in Biomedicine 13, 976–983. doi:10.1109/TITB.2009.2025373.

Pérez, J.S., Meinhardt-Llopis, E., Facciolo, G., 2013. Tv-l1 optical flow estimation. Image Processing On Line 2013, 137–150.

Pichette, J., Laurence, A., Angulo, L., Lesage, F., Bouthillier, A., Nguyen, D.K., Leblond, F., 2016. Intraoperative video-rate hemodynamic response assessment in human cortex using snapshot hyperspectral optical imaging. Neurophotonics 3, 045003–045003.

Raabe, A., Van De Ville, D., Leutenegger, M., Szelényi, A., Hattingen, E., Gerlach, R., Seifert, V., Hauger, C., Lopez, A., Leitgeb, R., et al., 2009. Laser doppler imaging for intraoperative human brain mapping. NeuroImage 44, 1284–1289.

Ricco, S., Tomasi, C., 2012. Dense lagrangian motion estimation with occlusions, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE. pp. 1800–1807.

Richa, R., Bó, A.P., Poignet, P., 2011. Towards robust 3d visual tracking for motion compensation in beating heart surgery. Medical Image Analysis 15, 302–315.

Richards, L.M., Weber, E.L., Parthasarathy, A.B., Kappeler, K.L., Fox, D.J., Dunn, A.K., 2012. Intraoperative laser speckle contrast imaging for monitoring cerebral blood flow: results from a 10-patient pilot study, in: SPIE BiOS, International Society for Optics and Photonics. pp. 82074L–82074L.

Roberts, R., Potthast, C., Dellaert, F., 2009. Learning general optical flow subspaces for egomotion estimation and detection of motion anomalies, in: Computer Vision and Pattern Recognition, pp. 57–64.

Sdika, M., 2008. A fast nonrigid image registration with constraints on the Jacobian using large scale constrained optimization. Medical Imaging, IEEE Transactions on 27, 271–281. doi:10.1109/TMI.2007.905820.

Sdika, M., 2013. A sharp sufficient condition for b-spline vector field invertibility. application to diffeomorphic registration and interslice interpolation. SIAM Journal on Imaging Sciences 6, 2236–2257.

Sdika, M., Alston, L., Mahieu-Williame, L., Guyotat, J., Rousseau, D., Montcel, B., 2016. Robust real time motion compensation for intraoperative video processing during neurosurgery, in: Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on, IEEE.

Sobottka, S.B., Meyer, T., Kirsch, M., Koch, E., Steinmeier, R., Morgenstern, U., Schackert, G., 2013. Intraoperative optical imaging of intrinsic signals: a reliable method for visualizing stimulated functional brain areas during surgery: clinical article. Journal of neurosurgery 119, 853–863.

Spinczyk, D., Karwan, A., Copik, M., 2014. Methods for abdominal respiratory motion tracking. Computer Aided Surgery 19, 34–47. doi:10.3109/10929088.2014.891657. pMID: 24720494.

Steimers, A., Gramer, M., Ebert, B., Fchtemeier, M., Royl, G., Leithner, C., Dreier, J.P., Lindauer, U., Kohl-Bareis, M., 2009. Imaging of cortical haemoglobin concentration with RGB reflectometry, in: European Conference on Biomedical Optics, Optical Society of America. p. 7368_13.

Szeliski, R., 2010. Computer vision: algorithms and applications. Springer Science & Business Media.

Thirion, J.P., 1998. Image matching as a diffusion process: an analogy with maxwell's demons. Medical image analysis 2, 243–260.

Villringer, A., Chance, B., 1997. Non-invasive optical spectroscopy and imaging of human brain function. Trends in neurosciences 20, 435–442.

Wulff, J., Black, M.J., 2015. Efficient sparse-to-dense optical flow estimation using a learned basis and layers, in: IEEE Conf. on Computer Vision and Pattern Recognition.