Inverse Formulation of Temporal Closure and Proposed Solutions for Offset Tracking of Natural Scenes

Héla Hadhri, Emmanuel Trouvé, Abdourrahmane Atto, Flavien Vernier

Inverse Formulation of Temporal Closure and Proposed Solutions for Offset Tracking of Natural Scenes

H. Hadhri*, E. Trouvé*, A. Atto*, and F. Vernier*
* Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance
Polytech Annecy-Chambéry, Université Savoie Mont Blanc.
Email: hela.hadhri - emmanuel.trouve - abdourrahmane.atto - flavien.vernier@univ-smb.fr

***Abstract*—Object tracking from time-lapse images or videos is a well-studied problem. Solutions include parameterized motion models and optical flow computing. It is necessary though, in some contexts, to further regularize the resulting tracking measurements. In this paper, we propose an approach based on temporal coherence constraints formulated as an inverse problem. Two formulations of the problem are established. These formulations are based on the construction of a network of couples within which the deformation must be coherent in order to get rid of all the artifacts induced by the measurement process in natural environments. Evaluation is done on a simulated dataset of glaciers scenes and experimental results are presented on time series acquired by ground-based cameras used for the monitoring of Alpine glaciers.** [1]

I. Introduction

In the current scenarios of climate change ambiguity, glacier shrinkage is recognized as a high-confident climate indicator as glaciers have been labeled an "Essential Climate Variable" [**?**] requiring systematic observation and average annual melting rate of mountain glaciers appears to have doubled after the turn of the millennium.

Such rapid environmental changes require that the glacier monitoring efforts make use of the swiftly developing new technologies. Modern ice investigation and forecasting are therefore being done by satellite imaging and increasingly by optical imaging as it turns out one of the very best applications of artificial intelligence is computer vision.

An intelligent processing of a glacier scene time-lapse would be to accurately track an object on the scene independently of missing frames, scene obstructions or of any eventual season or point of view switch during the observation time.

Although over the last ten years the computer vision field has acquired a significant methodological maturity and has been used successfully in a wide range of applications, usually industrial, urban or for laboratory experiments, it is still suffering from a great dependence on the conditions of acquisition and observational noise.

Computer vision algorithms, thus, require further constrainess to be applied to the observation of natural environments whose dynamic is less deterministic and whose observational noise is unpredictable, especially for non-supervised processing.

In this paper, we present the methodology we developed for change detection and offset tracking over alpine glaciers by automatic analysis of time-lapses acquired with a ground based camera.

We briefly introduce all the preprocessing steps applied to the time lapse: Spatial information is first evaluated to discard useless data and prepare relevant one. Parametrized deformation models are then used to compensate camera motion and compute unbiased optical flow fields of the observed object between different dates. The last stage of the chain, the tracking step that regularizes the said motion measures by constraining them with temporal closure, is the main subject of this paper.

The temporal closure here is formulated as a linear mapping from the model space, containing the tracking offsets, to the data space, containing the measured or observed offsets. Two formulations of this mapping are proposed and tracking offsets maximizing the temporal closure constraints are sought by inversion.

The proposed algorithm is fully automatic and takes as Input a link to a full time-lapse even if some of the images were not correctly acquired. It gives as output the selected and co-registered images, various intermediate scores, images and indexes, the matching measures and the displacement path through the time-lapse of a chosen pixel or image region.

These tracking results are useful for our collaboration with geomorphologists and important for constructing a consistent time series for future research.

II. Preprocessing steps

Constructing an accurate image time-series requires to first select the useful images on valid dates. Images acquired at the same time on a periodic time scale are first selected by dint of their meta-data files. The relevant images are the ones containing most textural information. Textural information is quantified by means of the amount of spatial gradients that we flatten onto a score. Considering that these scores follow a Gaussian distribution, the images to be rejected are those

[1]This work was supported by PHOENIX ANR-15-CE23-0012 grant of the French *National Agency of Research*.

978-1-5386-3327-4/17/$31.00

whose score constitutes a value considered aberrant, according to the Chauvenet test.

The most suitable geometrical transformation for modeling this change of perspective is homography, which models the perspective projection of a plane in a 3D frame, and thus takes into account the fact that the camera can also "lean" contrary to an affine transformation that would have modeled a change in perspective, but in the same 2D plane. This co-registration is done only on the parts of the scene that are supposed to be static, *i.e.*, the glacier is being masked whenever segmentation is trivial (some glaciers are the only white objects during hot seasons) and the mask is roughly propagated to the other images (by making it cover $\sim 20\%$ more than the glacier zone).

At this stage, we theoretically got rid of most of the non-homogeneous bias induced by our measurement system and we are able to estimate the glacier motion. For this sake, first, dense correspondences between all the images are computed [**?**]. This second "registration" step assume a local and non-rigid deformation model and can therefore efficiently determine dense correspondences in the presence of significant geometric and photometric changes between images. The dense matching uses a quad-tree like scheme to compute a pyramid of hierarchical, multi-layer correlation maps; local matches are computed assuming a restricted set of feasible deformations and are then propagated up to the hierarchy, which progressively discard spurious incorrect matches by down-sampling, applying a max-pooling and a non-linear power transform. This way, correlation maps of small patches (Base of the pyramid) are aggregated to form correlation maps of bigger patches. This alternation of aggregation between channels from the previous layer with channel-wise max-pooling, subsampling and a non-linear rectification makes this scheme similar to Convolutional Neural Networks but instead of learning the weights of the convolutions it computes the correlation pyramid. Retrieval of atomic 4x4 patches is then done by backtracking the dynamic of increasingly smaller patches from the top of this pyramid.

This dense matching makes further analysis weakly affected by lens distortions and shows robustness against repetitive texture. The parameters are tuned as to restrict the search area of a match to a known range that cannot be exceeded.

We then use these dense matches to initialize a further computation of a regularized dense optical flow field [**?**] $\vec{F}$ that minimizes an energy functional based on a data term, from the brightness constancy assumption and the gradient constancy assumption, and a regularizer, encouraging smoothness of the flow field.

The parameters of the optical flow computation are tuned as to favor the gradient constancy assumption over the brightness constancy assumption. This makes the optical flow computation more sturdy to brightness, weather and season switch.

To quantify the preprocessing step, various scores and error measures are computed, including:

- The spatial gradients score;
- The residual homography from the dense matchings using the Direct Linear Transform algorithm [**?**];
- A correlation score per couple of images equal to the sum of all patch similarities along their back-tracking path;
- The average residual displacement in the static regions;
- The mean flow direction on different parts of the glacier by the means of circular random variables statistics and their coherence score according to prior knowledge on the glacier motion (incoherent directions, like retreating);
- The temporal closure error map within a chosen time range.

III. Regularization by inversion

A. State of the art

Inversion consists in inferring the manifold of structural models that could yield some actual observations through a given transformation induced by a physical theory. It was first studied for solving the Sturm–Liouville theory in 1929. Since that time, the algebra tool is used extensively in many fields like earth observation [**?**] [**?**] [**?**], computer vision [**?**] or machine learning to infer data that cannot be directly observed. The physical theory that our inversion is based on is simply the consistency of the displacement pattern over time: the temporal closure constraint.

The temporal mis-closure concept was first introduced in Interferometric synthetic aperture radar technologies as an error to minimize for phase unwrapping of interferograms [**?**] and further work on the regularization of the temporal closure constraint has been conducted [**?**] for earth deformation monitoring.

In this paper, besides the temporal mis-closure introduced by the state of the art, we introduce a second formulation based on the same temporal coherence concept.

B. Initial problem and possible formulations of the temporal closure constraint

Classic time-lapse analysis usually computes offsets only between successive dates. The proposed method's aim to not limit the time lapse analysis on consecutive dates but to construct a network of homologous date indexes in between which displacement fields are computed. This scheme enables to check for and constrain by displacement temporal coherence.

The displacement fields measured between all possible date index arrangements, backwards and forward, form the observations. The unknowns to be solved for by inversion, or in this case reestimate, are the tracking offsets, meaning the displacements on forward consecutive date indexes. These are the estimates. Figure 1 schematizes the indexes of the dates in between which we computed an optical flow field with an arrow going from master image date index to the slave image date index.

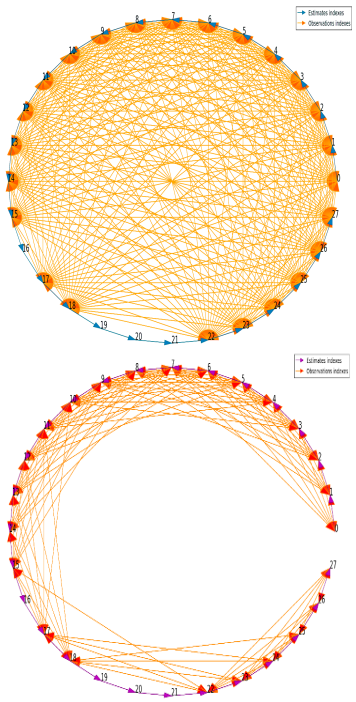

Fig. 1. TOP: Graph representation of the indexes of the dates in between which optical flow is estimated. The vertices of the polygon are the date indexes. The blue arrowed edges form the successive date indexes of the estimates, or tracking offsets, and the orange arrows are the date indexes of the observations. We can see that these are double-headed arrows between all possible indexes meaning flow fields were computed between all possible arrangements of dates. Images with date indexes 16, 19 and 21 are images that have been discarded on the preprocessing steps for lack of information. BOTTOM: Graph for a reduced range system. In this case $a = 7$.

The a priori information enabling us to establish a relation between the observations and the estimates can be expressed in 2 ways:

$$\forall\ i \text{ in Image Indexes} : \sum_{i=a}^{a+n-1} \vec{F}_{i,i+1} - \vec{F}_{a,a+n} = \vec{0}, \quad (1)$$

$$\forall (i,j) \text{ in Image Indexes} : \quad \vec{F}_{i,j} + \vec{F}_{a,i} - \vec{F}_{a,j} = \vec{0}. \quad (2)$$

$\vec{F}$ denotes the optical flow vector in a given pixel, a and n are respectively the master image index and the temporal closure range, meaning the number of adjacent optical flow fields involved in the problem. We solve these equations for the horizontal and vertical components separately and for every pixel in parallel.

These linear mappings can be expressed by the means of linear matricial transformations $A \times F_{est} = F_{obs}$ where F_{obs} are the optical flow components F_{x-obs} and F_{y-obs} of the observations belonging to the Data space $\mathbb{D}$ and F_{est} are those of the estimates belonging to the model space $\mathbb{M}$.

To illustrate this; for a time series of 28 observations, a full range of the temporal closure and the master image being the image with index 0, the obtained linear transformations are respectively:

$$\begin{pmatrix} F_{obs0-1} \\ F_{obs1-0} \\ F_{obs0-2} \\ F_{obs2-0} \\ \vdots \\ F_{obs0-27} \\ \vdots \\ F_{obs25-27} \\ F_{obs27-25} \\ F_{obs26-27} \\ F_{obs27-26} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ -1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & \dots & 0 & 0 \\ -1 & -1 & 0 & \dots & 0 & 0 \\ & & & \vdots & & \\ 1 & 1 & 1 & \dots & 1 & 1 \\ & & & \vdots & & \\ 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & -1 & -1 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & -1 \end{pmatrix} \begin{pmatrix} F_{est0-1} \\ F_{est1-2} \\ F_{est2-3} \\ \vdots \\ F_{est25-26} \\ F_{est26-27} \end{pmatrix} \quad (3)$$

$$\begin{pmatrix} F_{obs0-1} \\ F_{obs1-0} \\ F_{obs0-2} \\ F_{obs2-0} \\ \vdots \\ F_{obs0-27} \\ \vdots \\ F_{obs25-27} \\ F_{obs27-25} \\ F_{obs26-27} \\ F_{obs27-26} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ -1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 0 & \dots & 0 & 0 \\ & & & \vdots & & \\ 0 & 0 & 0 & 0 & 0 & 1 \\ & & & \vdots & & \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} F_{est0-1} \\ F_{est0-2} \\ F_{est0-3} \\ \vdots \\ F_{est0-25} \\ F_{est0-26} \\ F_{est0-27} \end{pmatrix} \quad (4)$$

These matrices can be implemented algorithmically as follows:

Algorithm 1: Constructing the linear transformation constraining the temporal closure: Formulation 1

Require: d: number of observed (valid) dates, d_{tot}: total range of dates

$A \leftarrow$ zeros($d \times (d-1)$, $d_{tot} - 1$) {Initializing matrix}

$\ell \leftarrow 0$ {Initializing the counter for lines of the matrix}

for (i,j) in **arr**(d_{tot},2) **do**

  if $i < j$ **then**

    A[ℓ, i:j] = 1

  else

    A[ℓ, i:j] = -1

  end if

  $\ell \leftarrow \ell + 1$

end for

Algorithm 2: Constructing the linear transformation constraining the temporal closure: Formulation 2

Require: d: number of observed (valid) dates, d_{tot}: total range of dates

$A \leftarrow$ zeros($d \times (d-1)$, $d_{tot} - 1$) {Initializing matrix}

$\ell \leftarrow 0$ {Initializing the counter for the lines of the matrix}

for (i,j) in **arr**(d_{tot},2) **do**

  $A[\ell, i] = -1$

  $A[\ell, j] = 1$

  $\ell \leftarrow \ell + 1$

end for

Where **arr** is the function that gives all possible arrangements of index couples (second argument) within the range of the total number of date indexes (first argument).

We can see that the estimates from inversion of the version encoding equation 1 and the ones from equation 2 are related by:

$$F_{est_2}(t) = \int F_{est_1}(t)dt, \tag{5}$$

C. Resolution

The system of equations expressing the temporal closure can be a full rank problem or rank deficient depending on the number of missing dates. The rows of A and F_{obs} corresponding to a couple of images that has a rejected index are deleted and hence some columns become linearly dependent reducing the rank of the, already smaller, matrix.

The problem to be solved is then:

$$find \quad \hat{F}_{est} \quad | \quad A \times \hat{F}_{est} = F_{obs} \tag{6}$$

In case of missing frames this problem is either inconsistent or has an infinite number of solutions; in some cases, A becomes fat (more columns than rows), the problem becomes over-determined and we look for the solution $\hat{F}_{est}$ that minimizes the squared L^2-norm of the error $\|\epsilon\|_2^2 = \|A \times F_{est} - F_{obs}\|_2^2$. In out case, most of the time, the number of rejected images is not large enough and the problem becomes under-determined. In this case, the set of all minimizers is convex and has a unique element having minimum length. And we are seeking this min length solution that *regularizes* our tracking observations. That is,

$$\begin{aligned} &\hat{F}_{est} = min\{\|F_{est}\|_2\} \\ &\text{subject to:} \\ &AF_{est} = F_{obs}, \end{aligned} \tag{7}$$

or more generally subject to:

$$A^T A F_{est} = A^T F_{obs} \tag{8}$$

where A can have any shape.

This can be solved by computing the solution to equation 8 that gives:

$$\hat{F}_{est} = (A^T A)^{-1} A^T F_{obs}, \tag{9}$$

or

$$\hat{F}_{est} = A^T (A A^T)^{-1} F_{obs}, \tag{10}$$

for the case of A becoming fat.

$A^\dagger = (A^T A)^{-1} A^T$ or $A^T (A A^T)^{-1}$ denotes the well-known Moore-Penrose pseudo-inverse that can be computed by means of a Cholesky factorization of either $A^T A$ or $A A^T$. In the over-determined case, this is the least expensive approach, but it is also the least accurate, especially on ill-conditioned problems as $A^T A$ and $A A^T$ are not invertible if A is not full rank.

Another way to solve for equation 7 is by performing an orthogonal factorization of A, for example, by SVD decomposition as follows:

Algorithm 3: Solving $A \times F_{est} = F_{obs}$ by Singular Value Decomposition

1: Find the SVD of $A = UDV^T$
2: Set $b = U^T F_{obs}$ {The problem becomes minimizing $||Dy - b||$}
3: Find the vector y defined by $y_i = b_i / d_i$, where d_i is the i-th diagonal entry of D
Return $F_{est} = Vy$.

This way, for handling rank deficiency (and thus avoid dividing by 0 singular values d_i in step 3), y_i is set to 0 for $i > rank(A) = r$. The family of solutions is:

$$F_{est} = Vy + \lambda_{r+1} v_{r+1} + ... + \lambda_n v_n, \tag{11}$$

where $v_{r+1} ... v_n$ are the last $n - r$ columns of V.

D. Weighted and Generalized Least-Squares

Ordinary least-squares minimizes the squared error when the data are homoskedastic, *i.e.*, when the variance of the noise term $\epsilon \mid AF_{est} = F_{obs} + \epsilon$ is considered constant for all observations. The problem is considered heteroscedastic when ϵ varies around the data space $\mathbb{D}$.

In our case, the magnitude of the noise error ϵ depends on F_{obsi-j} and as errors are accumulated all the way through the processing chain it strongly depends on the difference between the two images. Far apart images in time (range of the indexes) or in similarity, due to meteorological changes, will show more error through the measurement process.

To tackle heteroskeadasticity we may *focus accuracy* by caring strongly about the response for certain values of the input than others or *discount imprecision* by using weights that are inversely proportional to the variances of the observations.

For both cases this is equivalent to preconditioning the matrix A or to minimize the weighted norm $\|.\|_W$ of ϵ where W defines the weights of the inner-product defined by $< a, b >_W = b^* W a = b^T W a$ and thus must be a positive-definite bilinear preferably symmetric form, but can be non-symmetric in some cases. As before, we can argue that the minimum error vector ϵ must be orthogonal in the inner product defined by W to the data space $\mathbb{D}$. This leads to the requirement $A^T W (AF_{est} - F_{obs}) = 0$ and rearranging it gives the weighted normal equation:

$$(A^T W A) F_{est} = A^T W F_{obs} \tag{12}$$

Practically, this can be implemented as follows:

Algorithm 4: Weighted Least-squares

Require: A, F_{obs}, W
1: $A_W = W \times A$
2: $F_W = F_{obs} \times W$
3: F_{est} = ResolveLS(A_W , F_W)

The weights have to be known (or more usually estimated) up to a proportionality constant.

1) Focusing accuracy: Here the weights are based on the prior stages for processing the observations. For every optical flow field F_{obsi-j} a score must be computed. This can be the average of a score computed for images i and j or any of the similarity scores listed in section II.

We chose to rely on the correlation score of the dense matches as the optical flow computation strongly relies on them.

In this case W is diagonal and these scores corresponds to the diagonal elements.

2) Discounting imprecision: To quantify the imprecision of every measurement one would want to use an estimator that captures the information about how volatile the entry of F_{obs} are.

This estimator can be the inverse of the variance or another measurement uncertainty. Since each weight is inversely proportional to the error variance, it reflects the quantity of information in that observation.

In our case, to estimate the error induced by the measurement process we rely on the static regions of our scene where the estimated offsets *should* be equal to zero:

$$\begin{cases} F_{obs} = AF_{est} + \epsilon \\ F_{est} = 0 \end{cases} \Rightarrow F_{obs} = \epsilon \tag{13}$$

As already mentioned, these static regions are known by segmentation, when it's trivial, or by providing a manual binary mask of the scene (static / dynamic regions) with the inputs of the automatic processing chain. The variance is then computed on a spatial static neighborhood. Assuming that our optical flow field should be homogeneous, this variance denotes a measure uncertainty.

Another way is to first compute one or multiple non-weighted least-squares solutions from which we derive the residuals, estimate their correlation matrix and inverse it.

The distance $\epsilon^T W \epsilon$ is the Mahalanobis distance. The case when there exists a correlation between the residuals is the case when the weight matrix W is no longer diagonal and the solution is then the Generalized Least-Squares solution.

3) Iterative solving: The problem to inverse involves a large quantity of data, $A^T A$ have a high condition number and $A^T W A$ may be severely ill-conditioned, depending on the weight matrix.

The obtained solution is thus very sensitive to perturbations which may stem from observational noise, errors in the data, round-off errors, discretization errors, etc.

Using an iterative scheme to obtain accurate solutions is a way to deal with these difficulties.

The used method: Sparse Equations and Least Squares (LSQR) [**?**] is an iterative method which is analytically equivalent to performing a *Conjugate-Gradient* (CG) resolution scheme for problem 7 on which we added a term for a *linear regularization*.

Conjugate-Gradient method: For solving $Ax = b$, where A is symmetric and positive-definite, the iterative CG method looks for a set of mutually-conjugate vectors $\{p_1, \cdots, p_n\}$ that form a basis P and computes the coefficient of x, $\alpha_i \mid x = \sum_i^n \alpha_i p_i$, in that basis at each iteration i. The solution is thus gradually built along the dimensions of the solution space.

According to this scheme, we are minimizing the quadratic function $f : x \mapsto \frac{1}{2}(\mathbf{A}x, x) - (F_{obs}, x)$ which is equivalent to minimizing the residuals ϵ as $\nabla f(x) = Ax - b$.

First, an initial guess x_0 of the solution is assumed. The first direction p_0 is then the negative gradient of f at $x = x_0$, which is equal to the residuals $p_0 = r_0$ at iteration 0. At each iteration i, the new direction is the conjugate, according to A, of the previous direction, that is:

$$p_k = -r_k + \beta_k p_{k-1} \tag{14}$$

where β_k is found by imposing the condition of conjugality according to A that is:

$$\beta_k = \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}$$

Linear regularization: The linear regularization is done by introducing a damping parameter $\lambda > 0$ so the solution is "regularized" in the sense that a unique solution always exists, and $\|x\|$ is bounded. The problem becomes minimizing: $\|Ax - b\|_2^2 + \lambda^2 \|x\|_2^2$.

To sum up, LSQR consists of applying CG on a damped version of 8:

$$(A^T A + \lambda^2 I) F_{est} = A^T F_{obs} \tag{15}$$

Using CG makes LSQR suitable for sparse matrices and the linear regularization makes the problem solvable even if the rank of A is not full.

IV. Results

The experimentation is done on a time-lapse of the Argentiere glacier located in the Mont Blanc massif (France). The acquisition was carried out between 13/09/2013 and 17/10/2013 at the rate of one image per day. For this acquisition, the camera used is a Panasonic DMC-LX3 packaged to withstand the extreme climatic conditions of the alpine environment. It's powered by a solar panel and programmed to automatically acquire the time-lapse. The installation is at an altitude of 2631m and about 500m from the studied glacier. This way, the observed scene has a surface of about 1 km^2.

Fig. 2. Argentiere glacier - Massif of Mont Blanc

Among the computed scores and maps, the temporal closure error map, between 2 dates chosen by the user of the processing chain, has proven effective in detecting events that have occurred in the studied area. The red area in figure

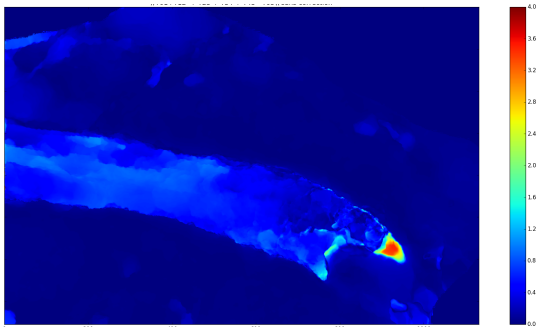

Fig. 3. Error map of the temporal closure constraint

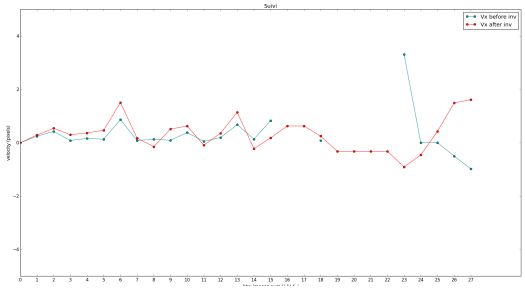

Fig. 4. Pixel tracking example: Observations VS estimated offsets.

3 corresponds to a Serac fall that have occurred during the chosen period.

We can also see in figure 4 that, apart from making the displacements coherent over time, the regularization extrapolates values on the missing indexes.

Another card generated for quantifying the glacier deformation is the mean flow over the time-lapse: figure 5. The mean amplitude and the main direction are encoded on the color space HSV where the Hue denotes the mean direction, computed by dint of circular random variables, the value denotes the amplitude of the flow and the saturation is set to 100%.

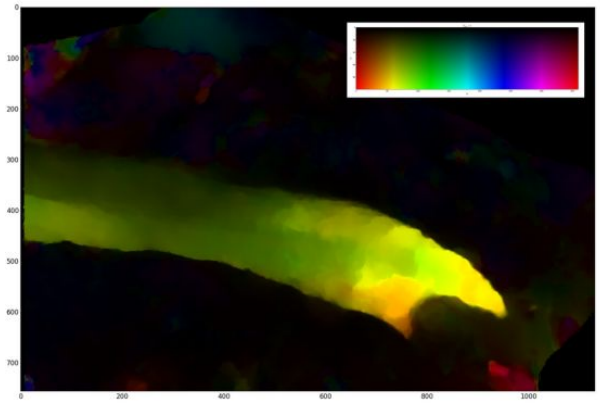

Fig. 5. Optical flow field on the Argentiere glacier - HSV color space.

The validation of the method has been conducted on a simulated dataset of a glacier scene in which the glacier deforms as a fluid and the background stays still. The obtained displacement map is then transformed by adding a random Gaussian noise on each pixel and on each date index to form the observations.

Simple, iterative weighted or non-weighted least squares resolutions are compared in figure 6.

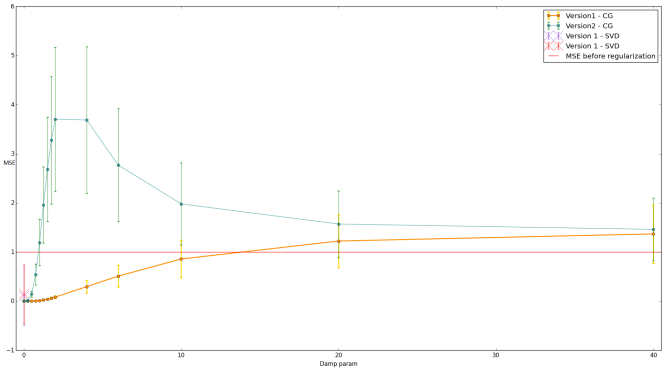

Fig. 6. Resolution parameter λ^2 in terms of the Mean Squared Error - Resolution by SVD: with version 1 and 2 of the temporal closure constraint.

The first formulation 3 is the one that have been validated because it has proven to be less sensitive to the damping parameter and to give better results.

We can see in figure 7 that the Least Squares and the Weighted Least Squares are both robust to outliers (Top) but that in some cases (Bottom), LS fails to reproduce acceleration and deceleration.

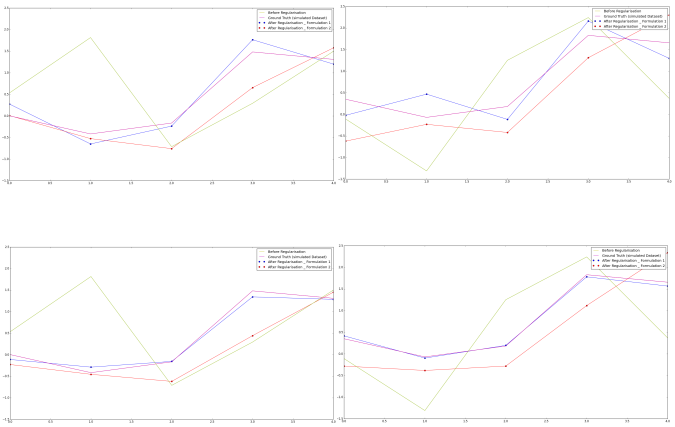

Fig. 7. Tracking examples - Top: Tracking pixel $(0, 50)$, Bottom: Tracking pixel $(200, 200)$, Left: Least-squares solution, Right: Weighted Least Squares solution.

Figure 8 shows the result of the regularization over all pixels. We can notice that the obtained regularized displacement map is a smoother, denoised version of the simulated observations. We also can see that the Weighted Least squares solution gives slightly better result.

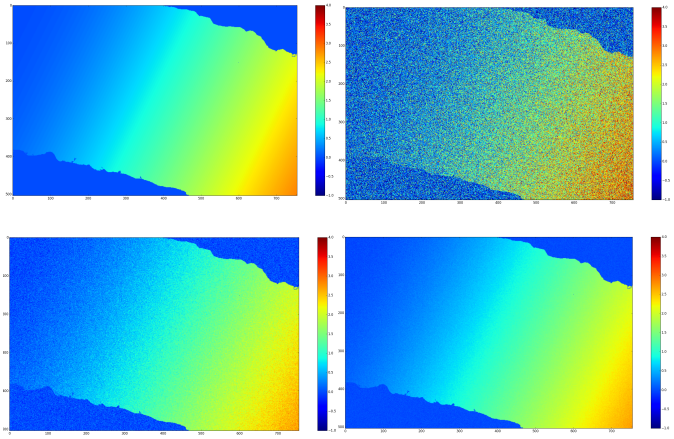

Fig. 8. x component of the regularized optical fields on the simulated data - Top left: Ground-truth, Top right: Noisy optical flow, Bottom left: regularized optical flow field with LS, Bottom right: Regularized optical flow field with WLS.

V. Conclusions and perspectives

This paper describes a technique for investigating surface deformations via time-lapses. After computing, as much as possible, unbiased optical flow fields between all the images, this approach maximizes the consistency of the deformation pattern over time by cross-checking at each date index on the consistency of the deformation offsets with respect to biases; it allows the easy combination of the optical flow fields computed via standard processing techniques and computes a regularized, coherent time sequence of the deformation. The technique we present is applied separately pixel by pixel to all the image and is robust with respect to possible errors of the preprocessing and optical flow computing steps. Since it is based on a simple inversion of a linear model, it is intrinsically possible to introduce further a priori knowledge about the temporal behavior of the deformation into the technique.

Even though the applied method is robust to outliers, it have shown lack of efficiency to regularize consecutive biased optical flow fields, as we can face, for example, when the range of the temporal closure constraint is within a hazy winter climate. In this case a lot of images gets rejected on the preprocessing step and the remaining optical fields can be biased.

In order to avoid problems related to images being too far away in the time-lapse, one would perform inversion on block-wise smaller systems given a range and a step value: figure 9.

An alternative to using iterative LSQR might be to build an iterative scheme for refining the mean and variance with jointly estimating the residuals by inversion and computing the new variances of the new residuals, thus using each to improve the other. This can be done by using squared residuals or log squared residual.

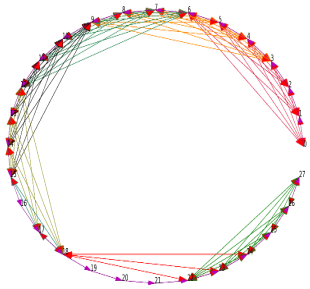

Fig. 9. BLock-wise system where every color is a system to be solved on its own. In this case $a = 6$ and the step value $= 3$.