A Proximal Algorithm for Joint Blood Flow Computation and Tissue Motion Compensation in Doppler Ultrafast Ultrasound Imaging

Nora Ouzir, Vassili Pustovalov, Duong-Hung Pham, Denis Kouamé, Jean-Christophe Pesquet

A Proximal Algorithm for Joint Blood Flow Computation and Tissue Motion Compensation in Doppler Ultrafast Ultrasound Imaging

N. Ouzir, *Member, IEEE,* V. Pustovalov, *Member, IEEE,* D. H. Pham, *Member, IEEE,* D. Kouamé, *Senior, IEEE,* and J.-C. Pesquet, *Fellow, IEEE,*

***Abstract*—Accurate tissue-clutter rejection and blood flow estimation remain challenging in ultrasound imaging. Traditionally, this estimation is performed by assuming static tissues. Only a few preprocessing techniques attempt to deal with the more realistic but challenging scenario where the tissues are moving. This paper tackles this scenario and presents a novel method for computing blood flow from moving tissues in ultrafast ultrasound imaging. The proposed computational ultrasound imaging method solves a global inverse problem that jointly computes blood flow, tissues, and their motions. The resulting cost function incorporates each component specificity using appropriate regularizations and is fully convex. The cost function is minimized using an alternating proximal-forward-backward algorithm with convergence guarantees. Furthermore, the proposed method is integrated into a multi-resolution scheme for large motions. The experiments show that the proposed method accurately compensates tissue motions, improving the precision of blood flow computation compared to previous methods. Experiments on *in vivo* images demonstrate the effectiveness of the proposed method in realistic scenarios with large motions.**

***Index Terms*—Blood Flow Computation, Motion Compensation, Brain Tumour, Doppler Ultrasound, Doppler ultrasonography, Ultrafast Ultrasound Imaging.**

I. Introduction

BRAIN surgery is the usual treatment for most brain cancers. A neurosurgeon partially opens the patient's skull during surgery, known as a *craniotomy*, to access the brain. The high-precision removal of the tumour then necessitates an accurate definition of the boundary between the tumour and other vital brain tissues. This assessment is usually done visually by the surgeon with the help of medical imaging techniques, such as ultrafast ultrasound imaging (UUI), which has recently become more widespread in clinical practice [1]–[3]. However, the precise demarcation of the boundary remains a complex task mainly because of the high vascularization of the *peritumoral* area, characterized by the proliferation of blood vessels, including tiny vessels with extremely low blood velocities [4]. Accurately determining the microvasculature of the *peritumoral* area through UUI-based high-sensitivity and high-resolution blood flow computation is thus gaining interest [5]–[10].

N. Ouzir and J.-C. Pesquet are with the Centre de Vision Numérique, Université Paris-Saclay, CentraleSupélec, Inria, 91190 Gif-sur-Yvette, France. E-mail: (nora.ouzir@centralesupelec.fr, jean-christophe@pesquet.eu).

V. Pustovalov, D. H. Pham and D. Kouamé are with the IRIT Lab, Université de Toulouse, and CNRS UMR 5505, 31400 Toulouse, France. Email: {vassili.pustovalov; duong-hung.pham; denis.kouame}@irit.fr.

UUI is a non-ionizing, low-cost, and real-time modality that non-invasively reveals anatomy, organ motions, or blood flow [11]. Unlike traditional ultrasound, UUI insonifies the whole region of interest using *plane* or *diverging* waves in a single ultrasound transmission, enabling imaging at significantly high frame rates (up to several thousands of Hz) [12], [13]. However, there is an inherent difficulty in computing blood flow in UUI and, more generally, ultrasound imaging. Because back-scattered blood signals (emanating from vessels) and clutter signals (from tissues surrounding the vessels) superimpose themselves on the images, distinguishing between blood and tissues is challenging, especially when the vessels are relatively small.

Blood flow computation methods deal with the issues above by removing unwanted clutter signals from stationary or moving tissues to reveal blood flow as clearly as possible. The traditional approach suppresses clutter using spatiotemporal filtering based on singular value decomposition (SVD). This simple strategy has led to numerous other techniques, such as the SVD of the correlation matrix of successive time-domain samples [14], a down-mixing method using an eigen-based tissue motion estimation [15], or a clutter rejection filtering method using an eigen-decomposition [16], to cite a few. Some years ago, [5] proposed a technique applying SVD to the images stacked in a 2D matrix (*i.e.*, the *Casorati* matrix). The smallest resulting singular values represent the noise, while the largest ones come from the tissues. Two thresholds are then empirically tuned to filter out these components and reveal the blood. This technique is fast and straightforward; however, the tuning process does not rely on any interpretable model [6]. More practical alternatives formulate blood flow computation as an inverse problem, where *a priori* knowledge about tissue and blood structures can be incorporated. Usually, the tissues are assumed to be low-rank, while the blood flow is sparse. This leads to a robust principal component analysis (RPCA) problem, typically solved by using iterative approaches such as the alternating direction method of multipliers (ADMM) [17]–[20]. Some variants of this method use a sparse regularization with a fixed basis [21] or a dictionary [22]. A considerable resolution improvement was achieved by incorporating a deconvolution step into the RPCA framework using the system point spread function (PSF). The PSF is experimentally measured or estimated jointly with the blood flow in BD-RPCA and its extension *fast* BD-RPCA [8]–[10]. Finally, a few techniques based on

deep learning have emerged recently [23], [24].

Previous methods have typically been developed based on the unrealistic assumption that tissues are static. In reality, motions always occur during data acquisition due to the ultrasound probe, the beating heart, breathing, or other unavoidable sources of patient movement. As a result, the correlation between frames decreases, undermining the assumption that tissues have a low-rank. This makes distinguishing between moving tissues and small blood vessels difficult, leading to inaccurate blood flow computation. Some works have addressed similar issues by incorporating a motion compensation strategy into applications combining UUI with other imaging techniques. In [25], a stable principal component pursuit (SPCP) was performed after applying a Lucas Kanade (LK) optical flow method, enabling a more accurate motion estimation. A super-resolution method also employed a two-stage motion correction procedure combining affine and nonrigid motions [26]. However, the methods mentioned above all use a standard motion estimation algorithm as a separate preprocessing step (*i.e.*, separated from blood flow computation), which may inadvertently compensate for the blood flow itself. Another major limitation of these approaches is the sequential estimation of tissue motions between frames (usually between the first frame - considered a reference - and all subsequent frames), leading to cumulative errors towards the end of the sequence.

This paper seeks to overcome the presence of motions by introducing a novel joint blood flow computation and tissue motion compensation method as part of a computational ultrasound system. The proposed approach computes blood flow simultaneously with the tissues and their motions, leading to more accurate blood and tissue separation. The approach is embedded in an inverse problem formulation using the sparsity and low-rank properties of the sought components. The proposed framework is also flexible as it can incorporate various motion regularizations. Specifically, we compare three classical regularization choices and derive the corresponding algorithms. Moreover, we carefully formulate a fully convex minimization problem in each case, solved using an alternating proximal forward-backward algorithm. More importantly, unlike the previous methods, the resulting algorithms have well-established theoretical foundations and convergence guarantees. Finally, the proposed method does not require the choice of a reference frame, which reduces the accumulation of motion errors.

The remainder of this paper is organized as follows. Section II presents the proposed joint blood flow, tissue and motion computation method. In Section III, we explain the proposed optimization strategy and provide the necessary details regarding implementation. Section IV shows experimental results using synthetic and real images. Finally, concluding remarks and perspectives are reported in Section V.

II. Joint Blood Flow, Tissue, and Motion Computation

This section presents the proposed problem formulation starting from the existing model without tissue motion estimation. The proposed model incorporating motion is then introduced and used to build a cost function to be minimized, as explained in the following subsections.

A. Problem Formulation in Previous Work

Let $S(\boldsymbol{y},t)$ be the observed complex analytic Doppler signal at time instant $t \in \{1,\dots,N_t\}$ and lateral/axial coordinates $\boldsymbol{y} = \begin{bmatrix} x \\ z \end{bmatrix}$ where $x \in \{1,\dots,N_x\}$ and $z \in \{1,\dots,N_z\}$. The signal S (recorded by a UUI system) is typically modelled at each spatiotemporal position as a superposition of tissues T, blood B, and additive noise N [9]:

$$S(\boldsymbol{y},t) = B(\boldsymbol{y},t) + T(\boldsymbol{y},t) + N(\boldsymbol{y},t). \tag{1}$$

After stacking the spatiotemporal 3D data into a 2D matrix (referred to as a Casorati matrix), Model (1) is usually reformulated for the whole sequence as

$$\boldsymbol{S} = \boldsymbol{B} + \boldsymbol{T} + \boldsymbol{N}, \tag{2}$$

where $\boldsymbol{S}, \boldsymbol{T}, \boldsymbol{B}$, and $\boldsymbol{N}$ are vectors in $\mathbb{C}^{N_x N_z \times N_t}$ corresponding to the stacked Casorati matrices of the observed data, tissues, blood, and noise, respectively.

In previous works, Model (1)–(2) has been used for different blood flow computation approaches, including SVD-based methods [14], [15] and inverse problem formulations such as RPCA [17], BD-RPCA [8], [9], and fast BD-RPCA [10]. The latter approach combines this model with constraints on the different matrices to formulate an optimization problem. However, Model (1)–(2) does not explicitly model tissue motions throughout the image sequence. In this work, our goal is to tackle blood flow estimation in the case of moving tissues, which can be due, for example, to image acquisition, patient motions, or the inherent motion of organs (*e.g.,* cardiac or respiratory motions). The following subsection reformulates the problem by including an explicit tissue motion component.

B. Inverse Problem Formulation with Tissue Motions

To incorporate motion into the previous model, we assume that the location of the sought blood B and tissues T should be constant throughout the image sequence. This means that the observed **moving** signal S is to be registered to a fixed grid of locations to retrieve the (ideally) non-moving components B and T. Note that the blood does move (*i.e., flow*) independently from the tissues inside the vessels. Here, we seek constant locations for the vessels while preserving these blood flow motions and compensating only for the underlying tissue displacements. Thus, by considering the displacement of the image sequence S with respect to A (an unknown registered version of the sought blood B and tissues T), we have

$$\begin{array}{ll} A(\boldsymbol{y},t) & = T(\boldsymbol{y},t) + B(\boldsymbol{y},t) \\ S[\boldsymbol{y} - \boldsymbol{d}(\boldsymbol{y},t),t] & = A(\boldsymbol{y},t) + N(\boldsymbol{y},t) \end{array}, \tag{3}$$

where $\boldsymbol{d}(\boldsymbol{y},t)$ is the 2D displacement vector, corresponding to the tissue motion at position $\boldsymbol{y}$ and time instant t, that is $\boldsymbol{d}(\boldsymbol{y},t) = [d_x(\boldsymbol{y},t), d_z(\boldsymbol{y},t)]^T$, with d_x and d_z the lateral

and axial motion components, respectively. Let us emphasize that in (3), the sought variable A acts itself as a reference for the motions $\boldsymbol{d}$. Similarly to the traditional optical flow approach [27], [28], we deal with the non-linearity induced by the shift in the pixel positions, by linearizing the left-hand side of (3) using a first-order Taylor expansion, that is

$$S[\boldsymbol{y} - \boldsymbol{d}(\boldsymbol{y},t),t] \simeq S(\boldsymbol{y},t) - \boldsymbol{\nabla} S(\boldsymbol{y},t)^\top \boldsymbol{d}(\boldsymbol{y},t), \quad (4)$$

where $\boldsymbol{\nabla}$ is a linear operator computing the spatial gradient in the lateral and axial directions. Since this linearization is only valid if the displacements are small, we will introduce a multiresolution approach in Section III-C to deal with large displacements and circumvent this limitation.

Then, by combining (3) and (4), the new model with tissue motions is

$$S(\boldsymbol{y},t) = A(\boldsymbol{y},t) + \boldsymbol{\nabla} S(\boldsymbol{y},t)^\top \boldsymbol{d}(\boldsymbol{y},t) + N(\boldsymbol{y},t). \quad (5)$$

In the following, (5) will be used to formulate an optimization problem allowing us to recover the different components B, T, and $\boldsymbol{d}$.

C. Proposed Cost Function

Using Model (5), uncovering blood flow from a moving image sequence requires jointly estimating the blood B, tissues T, and motion field d. The corresponding inverse problem can be solved by minimizing a cost function composed of a data fidelity term and appropriate constraints that can take the form of additional regularization terms (similarly to the RPCA method [17]) or introduce potential bounds on the values of the unknown variables.

1) Data Fidelity Term: Let subscript t index the vectorized variables corresponding to frame t, that is $\boldsymbol{S}_t \in \mathbb{C}^{N_x N_z}$ and $\boldsymbol{A}_t \in \mathbb{C}^{N_x N_z}$ designate the t-th columns of the Casorati matrices $\boldsymbol{S}$ and $\boldsymbol{A}$, and $\boldsymbol{d}_t \in \mathbb{R}^{2N_x N_z}$ is the 2D motion field of frame t. Using our new Model (5) and assuming an additive Gaussian noise N, we obtain the following data fidelity term:

$$h(\boldsymbol{A},\boldsymbol{d}) = \sum_{t=1}^{N_t} \|\boldsymbol{S}_t - \boldsymbol{A}_t + \mathcal{D}(\boldsymbol{d}_t)\boldsymbol{G}\boldsymbol{S}_t\|_2^2, \quad (6)$$

where $\boldsymbol{G}$ is the concatenation of the lateral and axial spatial discrete gradient operators, such that $\boldsymbol{G} = (\boldsymbol{G}_x, \boldsymbol{G}_z)^T$, with $\boldsymbol{G}_x \in \mathbb{R}^{N_x N_z \times N_x N_z}$ and $\boldsymbol{G}_z \in \mathbb{R}^{N_x N_z \times N_x N_z}$, and $\mathcal{D}$ is a linear operator defined as

$$\mathcal{D}(\boldsymbol{d}_t) = \begin{bmatrix} \mathrm{Diag}\left(d_x(t)\right) & \mathrm{Diag}\left(d_z(t)\right) \end{bmatrix} \in \mathbb{R}^{N_x N_z \times 2N_x N_z}, \quad (7)$$

with $d_x(t) = [d_x(\boldsymbol{y},t)]_{\boldsymbol{y}\in\mathcal{Y}}$ and $d_y(t) = [d_y(\boldsymbol{y},t)]_{\boldsymbol{y}\in\mathcal{Y}}$ where $\mathcal{Y} = \{1,\dots,N_x\} \times \{1,\dots,N_z\}$. Note that the data fidelity function in (6) has the advantage of being convex with respect to $\boldsymbol{A}$ and $\boldsymbol{d}$, and separable with respect to the temporal subscript t. It is also worth noting that the term $\boldsymbol{G}\boldsymbol{S}_t$ needs to be computed only once when evaluating it for various values of its input variables.

2) Constraints on T and B: The joint estimation of blood flow, tissue, and motion is a severely ill-posed problem. Therefore, it is necessary to introduce constraints that narrow down possible solutions. First, it has been shown in previous works that blood can be modelled as a sparse component while the tissue is expected to be low-rank due to its minimal change over time [7], [9], [17]. We follow the same strategy and add these assumptions to the minimization problem using the $\ell_{1,2}$-norm (*i.e.,* the sum of ℓ_2 norms of each row of its matrix argument, also denoted as $\|\cdot\|_{1,2}$) and nuclear norm (*i.e.,* the sum of the singular values of its argument denoted as $\|\cdot\|_*$). More precisely, the $\ell_{1,2}$-norm of $\boldsymbol{B}$ is defined as follows:

$$\|\boldsymbol{B}\|_{1,2} = \sum_{\boldsymbol{y}\in\mathcal{Y}} \Big[\sum_{t=1}^{N_t} |B(\boldsymbol{y},t)|^2\Big]^{1/2}. \quad (8)$$

Note that by using the above norm instead of the usual ℓ_1-norm [7], [9], [17], we have further considered that the blood locations tend to be constant over the sequence, *i.e.,* the motion compensated Casorati matrix $\boldsymbol{B}$ is sparse row-wise.[1] Finally, note that although the estimated tissue is not used in practice, it is required to estimate the blood in the iterations of the proposed method, as shown in Algorithm 1.

3) Constraints on $\boldsymbol{d}$: We also impose constraints on the unknown motion field $\boldsymbol{d}$, namely spatial smoothness combined with upper and lower bounds. Precisely, we introduce a box constraint limiting the estimated horizontal and vertical motions to the interval $[d_{\min}, d_{\max}]$. The interest of this box constraint is two-fold: (i) choosing appropriate bounds guarantees that the linearization (4) remains valid, and (ii) any *a priori* knowledge on the maximal magnitude of tissue motions can be considered (which can further aid the separation between relatively slow tissue motions and possibly faster blood flow). Spatial smoothness is a classical assumption typically incorporated using a Tikhonov or Total-Variation (TV) type regularization [27], [29], [30]. Both of these regularizations impose slower motion variations by acting on the motion gradients, which is consistent with clinical applications. To keep the proposed formulation flexible, we introduce a generic regularization term $\mathcal{R}(\boldsymbol{d})$ and propose to study the impact of both classical regularization choices. Note that we restrict our study to convex regularizations to preserve the convexity of the final cost function.

Let us first consider a smooth Tikhonov regularization acting separately on the horizontal and vertical components. In this case, the motion regularization term takes the form

$$\begin{aligned}\mathcal{R}_{\text{Tik}}(\boldsymbol{d}) &= \sum_{t=1}^{N_t} \|\boldsymbol{G}d_x(t)\|_{\text{F}}^2 + \|\boldsymbol{G}d_y(t)\|_{\text{F}}^2 \\ &= \sum_{t=1}^{N_t} \sum_{\boldsymbol{y}\in\mathcal{Y}} (\|\nabla_x \boldsymbol{d}(\boldsymbol{y},t)\|^2 + \|\nabla_z \boldsymbol{d}(\boldsymbol{y},t)\|^2), \end{aligned} \quad (9)$$

where ∇_x and ∇_z are the discrete gradient operators acting on the lateral and axial directions. $\|\cdot\|_{\text{F}}$ and $\|\cdot\|$ stand respectively

[1]The ℓ_1-norm regularization for the blood $\boldsymbol{B}$ has also been tested experimentally, but we have observed an improved performance using the $\ell_{1,2}$-norm.

for the Frobenius and ℓ_2-norms. Note that $\mathcal{R}_{\text{Tik}}(\boldsymbol{d})$ is the regularization used in the classic optical flow method [27]. It is characterized by a homogeneous smoothing of the motion field that can blur motion boundaries. Nonetheless, it has the advantage of being a smooth and quadratic convex term, which makes it easy to handle from a computational point of view. On the other hand, TV regularization performs edge-preserving smoothing. It can be expressed as

$$\mathcal{R}_{\text{TV}}(\boldsymbol{d}) = \sum_{t=1}^{N_t} \text{TV}_{\ell_p}(\boldsymbol{d}_t) \tag{10}$$

where, for every $t \in \{1, \dots, N_t\}$,

$$\text{TV}_{\ell_p}(\boldsymbol{d}_t) = \sum_{\boldsymbol{y}\in\mathcal{Y}} (\|\nabla_x \boldsymbol{d}(\boldsymbol{y}, t)\|_p + \|\nabla_z \boldsymbol{d}(\boldsymbol{y}, t)\|_p), \tag{11}$$

where $p \in [1, +\infty]$ and $\|\cdot\|_p$ denotes the componentwise ℓ_p-norm. For each pixel, (11) couples the different components of the displacement gradients, which results in better preservation of the discontinuities. In this work, we consider the two classical cases of an isotropic TV_{ℓ_2} ($p = 2$) and anisotropic TV_{ℓ_1} ($p = 1$) regularizations.

4) Final Cost Function: The data fidelity term (6) and different constraints lead to the following minimization problem:

$$\min_{\boldsymbol{B},\boldsymbol{T},\boldsymbol{d}} \sum_{t=1}^{N_t} \|\boldsymbol{S}_t + \mathcal{D}(\boldsymbol{d}_t)\boldsymbol{G}\boldsymbol{S}_t - \boldsymbol{A}_t\|_2^2 + \lambda\|\boldsymbol{B}\|_{1,2} + \mu\|\boldsymbol{T}\|_* + \rho\mathcal{R}(\boldsymbol{d}) + \sum_{t=1}^{N_t} \iota_{[d_{\min}, d_{\max}]^{2N_x N_z}}(\boldsymbol{d}_t), \tag{12}$$

where ι_E is the indicator function of a subset E of $\mathbb{R}^N$, equal to 0 if $x \in E$ and $+\infty$ otherwise, and λ, μ, and ρ are user-defined nonnegative real-valued parameters controlling the influence of each regularization term. The choice of these parameters will be discussed in more detail in Section IV dedicated to experiments. At this point, it is worth noting again that (12) does not require choosing a reference frame. Instead, it seeks the optimal transformation that aligns the images while satisfying the constraints (see Section IV). Let us also comment on some essential properties of the cost function in (12). The defined cost function is convex, proper, and lower-semicontinuous. In addition, it is coercive provided that $\lambda > 0$ and $\mu > 0$.[2] This guarantees the existence of a solution to the minimization problem.

Problem (12) can be efficiently solved by alternating minimization between the different blocks of variables $(\boldsymbol{B}, \boldsymbol{T})$ and $\boldsymbol{d}$. Because our problem combines convex smooth (h in (6) and $\mathcal{R}_{\text{Tik}}$) and non-smooth functions (the regularization on the blood/tissue and constraint on the motion), we resort to a proximal forward-backward algorithm. The next section explains this optimisation strategy choice and details the resulting algorithm.

[2]Function $f\colon \mathcal{H} \to]-\infty, +\infty]$ is coercive if $\lim_{\|x\|\to+\infty} f(x) = +\infty$, and it is proper if it is finite at least in one point. The reader is referred to [31] for more details on the class of lower-semicontinuous convex functions.

Algorithm 1: Alternating Proximal-Forward-Backward Algorithm for Joint Blood Flow, Tissue, and Motion Estimation Using a TV Regularization.

Initialization: Let $\boldsymbol{S} \in \mathbb{C}^{N_x N_z \times N_t}$, $\boldsymbol{T} \in \mathbb{C}^{N_x N_z \times N_t}$, $\boldsymbol{B} \in \mathbb{C}^{N_x N_z \times N_t}$ and $\boldsymbol{d} \in \mathbb{R}^{2N_x N_z \times N_t}$.
Set $\epsilon \in]0, 1]$.
for $k = 1, 2, \dots$ **do**
1) Let $\gamma_B^k \in [\epsilon, 2-\epsilon]$, set $\sigma_B^k = \gamma_B^k (L_d)^{-1}$:
- $\tilde{\boldsymbol{B}}^k = \boldsymbol{B}^k - \sigma_B^k \nabla_{\boldsymbol{B}} h(\boldsymbol{B}^k + \boldsymbol{T}^k, \boldsymbol{d}^k)$;
- $\boldsymbol{B}^{k+1} = \text{prox}_{\sigma_B^k \lambda\|\cdot\|_{1,2}}(\tilde{\boldsymbol{B}}^k)$ (*row-shrinkage*);

2) Let $\gamma_T^k \in [\epsilon, 2-\epsilon]$, set $\sigma_T^k = \gamma_T^k (L_d)^{-1}$:
- $\tilde{\boldsymbol{T}}^k = \boldsymbol{T}^k - \sigma_T^k \nabla_{\boldsymbol{T}} h(\boldsymbol{B}^{k+1} + \boldsymbol{T}^k, \boldsymbol{d}^k)$;
- $\boldsymbol{T}^{k+1} = \text{prox}_{\sigma_T^k \mu\|\cdot\|_*}(\tilde{\boldsymbol{T}}^k)$ (*soft thresholding of the singular values*);

3) Let $\gamma_d^k \in [\epsilon, 2-\epsilon]$, set $\sigma_d^k = \gamma_d^k (L_{B,T})^{-1}$:
- $\tilde{\boldsymbol{d}}^k = \boldsymbol{d}^k - \sigma_d^k \nabla_{\boldsymbol{d}} h(\boldsymbol{B}^{k+1} + \boldsymbol{T}^{k+1}, \boldsymbol{d}^k)$;
- $\boldsymbol{d}^{k+1} = \text{prox}_{\sigma_d^k \rho\mathcal{R}}(\tilde{\boldsymbol{d}}^k)$ (*see Algorithm 2*);

end
return *final* $\boldsymbol{B}, \boldsymbol{T}, \boldsymbol{d}$.

III. Alternating Proximal-Forward-Backward Algorithm

Proximal approaches provide a powerful framework for dealing with possibly non-smooth optimization problems arising in a high dimensional setting, such as processing image sequences [32], [33] containing hundreds of frames (*e.g.,* $N_t = 1000$ in our *in vivo* experiments). A further advantage of such algorithms is their capacity to handle problems involving high dimensional linear operators (*e.g.,* matrix $\boldsymbol{G}$) without requiring any matrix inversion. Because (12) involves three blocks of variables, we employ an alternating forward-backward algorithm, namely the Proximal Alternating Linearized Minimization (PALM) for non-convex and non-smooth problems [34].[3] These algorithms offer the possibility of using a different step size to update each of the variables, which results in an increased efficiency compared to approaches that are not block-coordinate-based. Finally, note that (12) is semi-algebraic (hence, satisfies the Kurdyka–Łojasiewicz (KL) property) and h is Lipschitz differentiable, which guarantees convergence to a solution to (12).

A. Algorithm

The proximal forward-backward approach alternates between gradient steps operated on the smooth part of the cost function and proximal steps handling the non-smooth constraints. In particular, the low-rank and sparse regularizations presented in Section II-C2 are convex but non-smooth functions, which makes the proximal approach helpful here. More specifically, the proximity operator of the

[3]The interested reader is referred to [35] for a recent extension of the PALM algorithm.

$\ell_{1,2}$-norm is obtained by computing the proximity operator of the Euclidean norm of each row of the argument matrix, while the proximity operator of the nuclear norm $\|\cdot\|_*$ performs a soft-thresholding on the singular values of the Casorati matrix (see Appendix B for precise expressions of the proximity operators). The motion regularization combining (10) (TV) and a box constraint also requires a proximal computation. In this case, the proximal operator does not have a closed-form expression but can be computed iteratively. The associated algorithm is presented in more detail in the following subsection.

Algorithm 1 summarizes the main steps of the proposed alternating proximal forward-backward algorithm for joint blood flow, tissue, and motion estimation in the case of a TV regularization. The operators $\nabla_{\boldsymbol{B}} h$, $\nabla_{\boldsymbol{T}} h$, and $\nabla_{\boldsymbol{d}} h$ denote the gradients of h with respect to $\boldsymbol{B}, \boldsymbol{T}$, and $\boldsymbol{d}$, respectively, L_d and $L_{B,T}$ are the Lipschitz constants associated with $\nabla_B h$ (or $\nabla_T h$) and $\nabla_d h$. Note that the constants γ_B^k, γ_T^k, and γ_d^k ensure that the steps satisfy $\inf_{k\in\mathbb{N}} \min\{\sigma_B^k, \sigma_T^k, \sigma_d^k\} > 0$ and $\sup_{k\in\mathbb{N}} \max\{L_d\sigma_B^k, L_d\sigma_T^k, L_{B,T}\sigma_d^k\} < 2$. Note also that the Tikhonov motion regularization (9) is convex and *smooth*. Therefore, it can be directly incorporated into the gradient step. The proximal step in stage 3 of Algorithm 1 then amounts to projecting onto the box constraint subset. For clarity, the associated modifications are enclosed in Appendix C.

B. Proximity Operator of $\mathcal{R}_{TV}$

The TV regularization (10) requires the computation of its proximity operator in step 3 of Algorithm 1. The proximity operator of the TV regularization does not have a closed-form solution and must be computed using an iterative approach. When considering an additional box constraint on the values of the motions and by using the separability of the involved regularization along the time dimension, the associated minimization problem can be expressed as

$$\mathrm{prox}_{\sigma_d^k \rho\mathcal{R}_{\mathrm{TV}}}(\tilde{\boldsymbol{d}}^k) \quad = \quad \left[\mathrm{prox}_{\sigma_d^k \rho \mathrm{TV}_{\ell_p}}(\tilde{\boldsymbol{d}}_t^k)\right]_{1\le t\le N_t}, \quad (13)$$

where, for every $t \in \{1, \ldots, N_t\}$,

$$\mathrm{prox}_{\sigma_d^k \rho \mathrm{TV}_{\ell_p}}(\tilde{\boldsymbol{d}}_t^k) = \underset{\boldsymbol{d}}{\mathrm{argmin}}\ \sigma_d^k \rho \mathrm{TV}_{\ell_p}(\boldsymbol{d}_t) + \iota_{[d_{\min}, d_{\max}]^{2N_x N_z}}(\boldsymbol{d}_t) + \frac{1}{2}\|\boldsymbol{d}_t - \tilde{\boldsymbol{d}}_t^k\|^2. \quad (14)$$

Notice that one has to solve N_t optimization problems simultaneously, which can be performed efficiently through parallel computation. Since the last two terms of the cost function in (14) define a strongly convex function, its Fenchel-Legendre conjugate is differentiable. This means that the dual optimization problem of (14) can be classically solved by a forward-backward algorithm [33], [36]. This approach is strongly related to the dual ascent algorithm widely used in machine learning applications [37]. In our case, an accelerated version of this algorithm is implemented where the FISTA-like approach in [38] is applied to the dual problem by introducing an iteration-dependent inertial parameter β_n. A summary of this approach is provided in Algorithm 2, where $p \in \{1, 2\}$, and proj_E denotes the projection onto a nonempty closed convex subset E of the Euclidean space.

Algorithm 2: Accelerated Dual Forward-Backward Algorithm for Computing $\mathrm{prox}_{(\sigma_d^k)^{-1}, \rho\mathcal{R}_{\mathrm{TV}}}$ for a TV Regularization with a Box Constraint.

Initialization: Let $\tilde{\boldsymbol{d}}^k, \boldsymbol{q}^k \in \mathbb{R}^{2N_x N_z \times N_t}$ be the motions and dual variables obtained after applying Algorithm 2 at iteration k of Algorithm 1, $\boldsymbol{r}_0 = \boldsymbol{q}_0$, $d_{\min} \in \mathbb{R}$ and $d_{\max} \in \mathbb{R}$.

Set $\epsilon_1 \in]0, +\infty[$, $\gamma_n \in]\epsilon_1, 1/\|\boldsymbol{\nabla}\|^2]$ and $\xi > 2$.

for $n = 1, 2, \ldots$ **do**
  for $t = 1, \ldots, N_t$ **do**
    $\boldsymbol{d}_{t,n+1}^{k+1} = \mathrm{proj}_{[d_{\min}, d_{\max}]^{2N_x N_z}}(\tilde{\boldsymbol{d}}_t^k - \boldsymbol{\nabla}^* \boldsymbol{r}_{t,n})$;
    $\tilde{\boldsymbol{q}}_{t,n} = \boldsymbol{r}_{t,n} + \gamma_n \boldsymbol{\nabla} \boldsymbol{d}_{t,n+1}^{k+1}$;
    $\boldsymbol{q}_{t,n+1} = \tilde{\boldsymbol{q}}_{t,n} - \gamma_n \mathrm{prox}_{\frac{\|\cdot\|_p}{\gamma_n}}(\frac{\tilde{\boldsymbol{q}}_{t,n}}{\gamma_n})$;
    $\beta_n = \frac{n}{n+1+\xi}$;
    $\boldsymbol{r}_{t,n+1} = \boldsymbol{q}_{t,n+1} + \beta_n(\boldsymbol{q}_{t,n+1} - \boldsymbol{q}_{t,n})$;
  end
end

$\boldsymbol{d}^{k+1} = \boldsymbol{d}_{n+1}$;

return *final* $\boldsymbol{d}^{k+1}$ *and* $\boldsymbol{q}^{k+1}$.

Algorithm 3: Multi-Resolution Approach for UUI Sequences with Large Motions.

Initialization: Let $J+1$ be the number of resolution levels, $\boldsymbol{d}$ the initial motions, and $\boldsymbol{S}$ the image sequence at the original resolution.

for $j = J, \ldots, 0$ **do**
  1) Warp the original size images with $\boldsymbol{d}$: $S_{\mathrm{warp}}(\boldsymbol{y}, t) = S[\boldsymbol{y} + \boldsymbol{d}(\boldsymbol{y}, t), t]$;
  2) Resize the warped sequence $\boldsymbol{S}_{\mathrm{warp}}$ to scale $\frac{1}{2^j}$;
  3) Obtain $\boldsymbol{T}^j, \boldsymbol{B}^j, \boldsymbol{d}^j$ using Algorithm 1;
  4) Resize variables $\boldsymbol{T}^j, \boldsymbol{B}^j$ to the next level scale $\frac{1}{2^{j-1}}$ (*if* $j \neq 0$, *for initializing step 3 at the next level*) ;
  5) Process motions for the next level
    - Resize $\boldsymbol{d}^j$ to original scale: $\boldsymbol{d}^j \leftarrow \mathrm{Resize}(\boldsymbol{d}^j)$
    - Re-scale motion magnitudes: $\boldsymbol{d}^j \leftarrow 2^j \boldsymbol{d}^j$
    - Increment the motions: $\boldsymbol{d} \leftarrow \boldsymbol{d} + \boldsymbol{d}^j$
end

return *Final* $\boldsymbol{B}$, $\boldsymbol{T}$, $\boldsymbol{d}$ *at the finest resolution level.*

C. Multi-resolution Approach for Large Motions

The motion of the ultrasound probe, compounded by tissue motions, can cause large displacements within the image sequence. However, the proposed method is designed for relatively small displacements (typically ensured by the bounds $d_{\max} = 1$ and $d_{\min} = -1$ also used in our experiments in Section IV). Therefore, we employ a multi-resolution approach

to deal with larger motions. Precisely, we use a pyramidal approach starting from a coarse resolution level J until the finest (original resolution) level 0 such that the scale at level j is $\frac{1}{2^j}$. Note that each level of the pyramid satisfies the box constraint defined by $d_{\max} = 1$ and $d_{\min} = -1$, while the final motion magnitude can be up to 2^J, where $J+1$ is the number of levels. It is important to note that this approach is designed so that the linearization in (4) remains valid across all pyramid levels (typically $J = 2$ to 5). At each level, the images are warped using the motions estimated from the previous level, gradually bringing the images to the final level, where the remaining motion updates are relatively small. More specifically, warping the images with the motions generated at a coarser level necessitates upsampling and rescaling of their magnitudes, *i.e.*, $\boldsymbol{d}^j = 2^j \mathrm{Resize}(\boldsymbol{d}^j)$, where the Resize function resizes the motions to the original image size. When distortions are too large, Gaussian filtering is typically used to reduce interpolation errors. The blood and tissue estimates can be propagated (by upsampling) from one level to another to speed up the computations, but the motions are always initialized with zero. A figure illustrating the proposed multi-resolution approach is provided as supplementary material. A detailed description of each step in the case of J levels is also provided in Algorithm 3.

IV. Experiments

This section evaluates the proposed method using synthetic and real data. We primarily focus on evaluating the quality of blood flow (the target variable) computation either quantitatively or visually. A comparison with different blood flow estimation approaches in terms of accuracy and computational efficiency is also presented. First, we consider the classical SVD technique [5] and the fBD-RPCA method [10].[4] It is helpful to recall that the fBD-RPCA method uses low-rank and sparsity constraints without assuming tissue motion. However, it includes a deconvolution step. In particular, it is interesting to see whether the deconvolution can compensate for a certain amount of motion in the sequence. Thus, the comparison with fBD-RPCA primarily focuses on showing the benefits of incorporating a joint motion estimation in the inverse problem approach. The second category of methods included in our analysis employs an explicit motion correction step: Piepenbrock [25], LK+RPCA [25], and Demon+fBD-RPCA [39] that will be referred to as *Demon*. All three approaches apply different blood computation techniques to the images after conducting a (separate) pre-processing step of motion compensation. More specifically, the first two methods use a multi-scale LK optical flow, while the third uses a Demon registration algorithm. Similarly to the proposed method, both Demon and multi-scale LK use a pyramidal multi-resolution approach to deal with large motions. Finally, we compare our results to a recent deep learning-based method, DeepfUS [23], which computes the power Doppler reconstruction of blood flow from sparse sequences of ultrasound data. This convolutional neural network was trained on in vivo US acquisitions of rat brains.

[4]fBD-RPCA is a fast extension of the BD-RPCA method whose improvements over RPCA have already been shown in [9].

A. Parameter Settings

In addition to choosing the regularization amount, the proposed method requires setting optimization and multi-resolution parameters. The following explains how to choose the parameters for each category. The parameters of state-of-the-art methods are also discussed, and an example of optimal hyperparameter values for the B_1 sequence is provided in Table I. The hyperparameters of all methods were tuned to maximize the performance measures used in the synthetic and *in vivo* experiments (see Sections IV-B2, IV-C2) using a grid search.

TABLE I
Optimal hyperparameters for the synthetic B_1 sequence.

Method	Parameters
SVD [5]	$T_c = 12$; $T_b = 14$
fBD-RPCA [10]	$T_c = 14$; $\lambda = 2$
LK+RPCA [25]	$\lambda = 1$; $w_1 = 3$; $w_2 = 5$; $w_3 = 9$
Piepenbrock [25]	$\lambda = 1.5$; $w_1 = 5$; $w_2 = 5$; $w_3 = 7$; $\lambda_2 = 0.01$; $\mu_2 = 300$
Demon [39]	$T_c = 1$; $\lambda = 1$; $n_{\mathrm{it}} = 5$; $w_{\text{demon}} = 10$; $\sigma_{\text{demon}} = 8$
Proposed $\mathcal{R}_{\text{Tik}}$	$\mu_0 = 1.7 \times 10^{-2}$; $\lambda_0 = 5.3$; $\rho = 0.5$
Proposed $\mathcal{R}_{\text{TV}-\ell_1}$	$\mu_0 = 1.7 \times 10^{-2}$; $\lambda_0 = 5.5$; $\rho = 1$
Proposed $\mathcal{R}_{\text{TV}-\ell_2}$	$\mu_0 = 1.7 \times 10^{-2}$; $\lambda_0 = 5.5$; $\rho = 1$

1) Regularization Parameters: For all methods using sparsity and low-rank constraints (including the proposed method), the regularization parameters are tuned according to the strategy proposed in [9]. First, the sparsity parameter is set to $\frac{\lambda_0}{\sqrt{\max(N_x N_z, N_t)}}$, where λ_0 is chosen in the interval $[10^{-1}, 10^2]$ to fine-tune a reference sparsity level. The low-rank parameter μ is also tuned using a grid search. To automatically adapt the parameters of the proposed method to the different pyramid levels (with different image sizes), we additionally follow the original RPCA strategy [17] where the regularization parameters are multiplied by the sequence norm (*i.e.,* $\mu = \mu_0 \|\boldsymbol{S}\|$ and $\lambda = \frac{\mu\,\lambda_0}{\sqrt{\max(N_x N_z, N_t)}}$). Finally, the spatial regularization parameter ρ is tuned separately for each regularization (*i.e.*, for each function $\mathcal{R}$) and motion field type: the smaller the ρ value, the more spatial variations are allowed. Thus, when the motions are spatially uniform, we simply choose a large value. Conversely, non-rigid motions (see Section IV-B) require smaller values. The final parameter values are detailed in Table I. For simplicity, low-rank and sparsity parameters of different methods are denoted similarly.

2) Optimization Parameters: For all data sets, the optimization parameters γ_T^k, γ_B^k, and γ_d^k are set to 1.95 to satisfy the convergence conditions while ensuring a sufficiently large step size. Similarly, the constant ξ in Algorithm 2 is set to 2.1. Algorithms 1 and 2 are stopped when the relative change of the blood flow is below 1e-3.

3) Multi-Resolution Parameters: All methods incorporating motion compensation in this work (Piepenbrock, LK+RPCA, Demon, and the proposed method) required more than 1 resolution level to obtain the best results. We used 6 pyramid levels in all experiments to ensure the proposed multi-resolution scheme can cover all motion magnitudes, especially in the case of unknown *in vivo* motions. As explained above, we did not tune each level regularization parameters separately since λ and μ change automatically with the image size and sequence norm. All other parameters remain constant across the different levels of the pyramid. The images were warped and resized using cubic interpolation (testing different interpolation types showed no significant impact on the results). For the synthetic sequence with the most severe spatial distortions (B_5), a Gaussian filter of size 5×5 and standard deviation of 1 was used before resizing.

4) Parameters Related to State-Of-The-Art Methods: In this subsection, we briefly explain the role of the remaining parameters in Table I (the reader is referred to the corresponding papers for a more detailed description of the methods). The parameters are tuned for all methods to maximize the performance of each image sequence. The SVD method [5] filters the eigenvalues using two thresholds: T_c for the tissues (or clutter) and T_b for the blood, such that $T_b < T_c$. LK+RPCA and Piepenbrock perform a motion correction step based on a multi-scale LK method [25], [39]. The LK method computes a constant optical flow in a local neighbourhood of size w_1. It also requires derivative and downsampling filter sizes w_2 and w_3. Piepenbrock also performs a low-rank decomposition of the motions, requiring additional sparsity and low-rank parameters denoted by λ_2 and μ_2, respectively.[5] Demon+fBD-RPCA performs a Demon registration before the fBD-RPCA algorithm. Consequently, three additional registration parameters are tuned: the number of iterations n_{it}, the size w_{demon}, and standard deviation σ_{demon} of the Gaussian kernel. Finally, we use the pre-trained network for DeepfUS [23], acknowledging the fact that we do not possess enough data for re-training. The neural network (U-net) structure takes input images of size 96×96 pixels. To process our sequences, which have larger sizes, we use overlapping patches (21 for each image in the synthetic sequences). The final power Doppler images are reconstructed from the patches and normalized.

B. Synthetic Experiments

Four synthetic ultrasound sequences with different tissue motions and a controlled ground truth were generated to quantitatively compare the performance of different methods.

1) Data Generation: The sequences contain 50 images of size 600×260 pixels. To simulate the RF signals, we followed the method of [40], [41], which is based on a convolution between the tissues and the system PSF.[6] The system impulse response was approximated by a spatially invariant PSF measured *in vitro* using laminar flow (blood-mimicking fluid) in a tube, where a sensitive Doppler acquisition was done with the SonicResearch interface Aixplorer equipped with an SL10-2 probe. Randomly positioned scatterers were drawn from a Gaussian distribution with a zero mean. To simplify the simulations, all tissue scatterers, including those at the borders of blood vessels, were drawn with the same standard deviation. The loss of amplitude due to reflections and absorption of US waves was also not taken into account. The scatterer grid spacing was 0.2 mm in both directions, and each resolution cell contained 5 scatterers.

For each sequence, two blood vessels were simulated by drawing separate scatterers. The blood flow was superimposed to the tissues at the location of blood vessels (*i.e.,* did not replace the underlying tissue). The blood flow scatterers had a standard deviation 5 times smaller than the tissue, leading to an amplitude ratio of -25dB [42]. To ensure a realistic ratio between the dynamics of the tissue and blood, the latter value was estimated from the *in vivo* data using a stationary echo cancelling method [43]. The motion of the blood inside the vessels was also tailored to the values of the *in vivo* sequences (image frame rate of 1kHz). All flow scatterers were moved inside each vessel with the same velocity and direction (constant velocity of 20 cm/s for the left vessel and 10 cm/s for the right one). As explained in the following, in addition to blood flow, synthetic tissue motions were generated and applied to the entire image, including the blood vessels. All motions were applied directly to the scatterers before direct convolution to obtain the final RF data.

To simulate moving tissues, two types of motion fields with different deformation magnitudes were applied to the entire images (*i.e.,* blood and tissues):

i) Random translations (mimicking motions of the probe) with a maximum magnitude of 0.2 mm and 1 mm (corresponding to 1 and 5 pixels) in both directions. The resulting sequences are denoted by T_1 and T_5, respectively.
ii) Non-rigid motions using B-splines (simulating the elastic motion of organs) with a maximum magnitude of 0.2 for the B_1 sequence and 1 mm for the B_5 sequence.

Note that the order of magnitude of tissue displacements due to cerebral pulsatility is 0.1 mm around the equilibrium position.

Finally, an additive Gaussian noise with an SNR of 35dB (where the signal is the RF data) was added to the images. Figure. 1 shows an example of the resulting images and motion fields in the case of B-splines motions with maximum displacements of 1 mm.

2) Results: The computed blood flow sequences are converted to power Doppler images. These images represent the power of the blood $\boldsymbol{B}$ (in dB) using all the sequence

[5]Because the Piepenbrock approach uses many parameters, λ_2 and μ_2 have been fine-tuned on one synthetic sequence only. Note that the influence of these parameters on the result is relatively small (less than half a dB for the PSNR), and the performance remained stable across different sequences.

[6]We have used a direct convolution instead of a grid approximation convolution to avoid motion artefacts. Furthermore, the study of out-of-plane motions is out of the scope of this work.

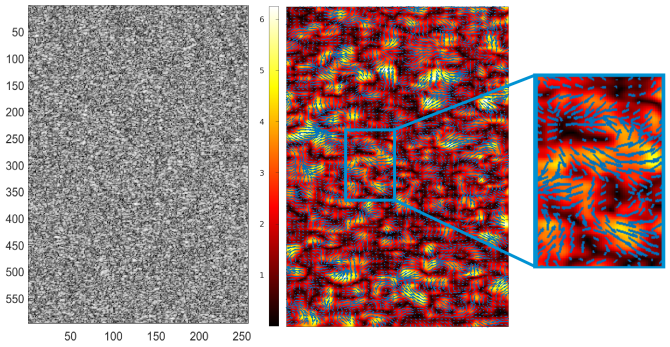

Fig. 1. (Left) a motionless synthetic image (intensities in the range $[0, 1]$). (Right) an example of simulated non-rigid tissue motions with their magnitude and direction (without blood flow).

TABLE II
NRMSEs (1ST ROW), PSNRs (2ND ROW), AND SSIMs IN % (3RD ROW) OF THE SYNTHETIC SEQUENCES WITH DIFFERENT MOTIONS.

Method	T_1	T_5	B_1	B_5
SVD [5]	0.37	0.54	0.37	0.45
	9.68	6.55	9.67	8.08
	3.59	2.04	2.76	2.06
Piepenbrock [25]	0.25	0.39	0.28	0.39
	13.1	9.36	12.3	9.23
	7.00	2.41	6.27	2.77
fBD-RPCA [10]	0.37	0.50	0.23	0.40
	9.73	7.19	13.9	9.03
	4.17	3.05	8.45	3.57
Demon [39]	0.32	0.54	0.23	0.35
	11.1	6.46	13.9	10.4
	7.01	3.46	7.98	4.66
DeepfUS [23]	0.43	0.44	0.22	0.37
	8.52	8.21	14.3	9.67
	3.48	3.30	9.05	3.82
LK+RPCA [25]	0.31	0.36	0.31	0.38
	11.4	9.96	11.3	9.56
	4.35	3.31	3.69	3.08
Proposed $\mathcal{R}_{\text{Tik}}$	**0.14**	**0.14**	**0.16**	**0.26**
	18.2	**18.1**	**16.9**	**12.7**
	61.0	57.4	**25.4**	**13.9**
Proposed $\mathcal{R}_{\text{TV}-\ell_1}$	**0.14**	0.15	0.17	0.28
	18.1	17.8	16.3	12.3
	60.8	**58.5**	23.9	11.6
Proposed $\mathcal{R}_{\text{TV}-\ell_2}$	**0.14**	0.15	0.17	0.28
	18.0	17.8	16.3	12.4
	60.8	58.5	23.9	10.5

frames. They are computed for each position $\boldsymbol{y}$ in the images as follows:

$$\boldsymbol{B}_{\text{PD}}(\boldsymbol{y}) = 10 \log_{10} \left(\frac{1}{N_t} \sum_{t=1}^{N_t} \boldsymbol{B}(\boldsymbol{y}, t)^2 \right). \tag{15}$$

The performance is then evaluated using the Normalized Root-Mean-Square Error (NRMSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM) computed on the power Doppler images [9], [23]. Note that for the synthetic sequences, the parameters are tuned to maximize the PSNR.

The obtained NRMSEs, PSNRs, and SSIMs are reported in Table. II. Generally, the performance decreases as the motions become larger. State-of-the-art methods failed at recovering blood flow for motions of 5-pixel magnitudes (T_5 and B_5). Moreover, some state-of-the-art methods using motion compensation resulted in significantly larger errors in both translation sequences compared to the B-spline-based motions. Note that these methods were designed for non-rigid motions. In contrast, the impact of motions is reduced for all versions of the proposed method, and the performance is superior for all motion types or magnitudes. In particular, there was no drop in performance in the case of translations. Recall that translations are spatially unvarying motions (mimicking probe motions) and can be recovered by a large enough regularization parameter value for all the proposed regularizations. Overall, the proposed method with the Tikhonov regularization performed slightly better than the TV-ℓ_1 and TV-ℓ_2 regularizations. A drop in performance was also observed for the proposed method in the case of non-rigid motions. However, the errors remained smaller than all other methods, and the blood flow was recovered successfully. Note that the decrease in SSIM between the T_1 and B_1 is due to the high sensitivity of this performance measure to noise, which was naturally higher for spatially varying motions where interpolation errors have more impact. Note also that considerably better SSIMs can be obtained by adding more sparsity, at the cost of slightly higher NRMSEs (*e.g.,* an SSIM of 60% can be also be obtained for the B_1 sequence with a 0.02 increase in NRMSE).

It is also interesting to compare the B_1 sequence errors: fBD-RPCA, which does not incorporate motion compensation, performed better than LK+RPCA for small motions. This could be due to an over-compensation of blood flows by the LK pre-processing step (applied before separating blood and tissues). In the absence of tissue motion (or for relatively small ones), it is likely that a separate correction step would compensate for blood flow and hinder its detection. In contrast, the proposed joint blood, tissue, and motion estimation mitigates this effect and preserves blood flow for small and large motions. Moreover, the state-of-the-art methods with a motion compensation step did not show a clear advantage compared to other methods for large motions (B_5 and T_5 sequences).

Fig. 2 shows the power Doppler images of the blood flows obtained for the B_1 sequence compared to ground truth[7]. Although our method succeeded for all sequences, B_1 was the only one for which all other methods revealed the blood flow. The figure shows that the proposed method preserved blood flow power while significantly reducing background noise compared to the other methods. We can also see that the proposed regularizations provided similar results with subtle differences. More noise reduction was obtained with Tikhonov regularization, which does not enforce motion boundaries and tends to recover a more global and smooth motion field. In contrast, the motion boundaries allowed by the TV regularizations can cause tiny amounts of motion compensation in the blood flow and at the boundaries between blood and tissues.

[7]A video showing the results of the estimated blood and tissues for the B_1 and T_5 sequences can be found at https://github.com/NoraOuzir/UUI-Blood-Flow-Motion.git

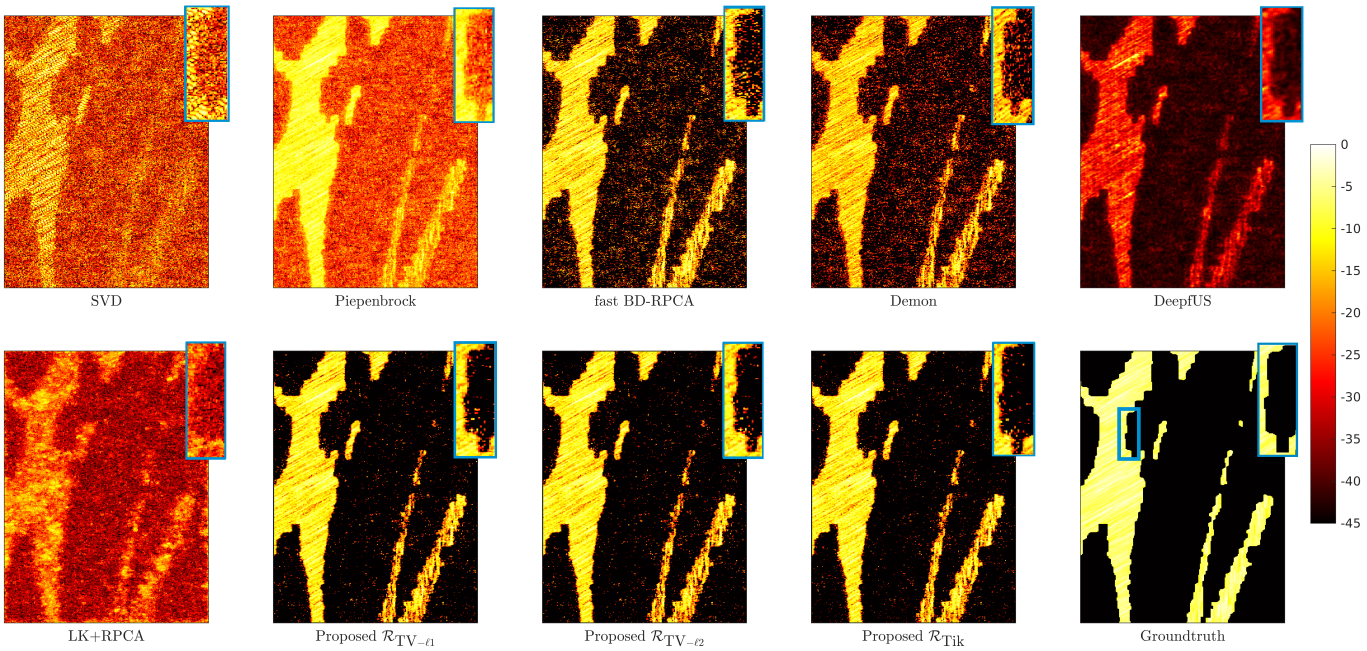

Fig. 2. Power Doppler images of the blood flows obtained for the synthetic B_1 sequence using: SVD; Piepenbrock; fBD-RPCA; Demon; DeepfUS; LK+RPCA; and the proposed method with different regularizations. All images are displayed with a dynamic range of 45 dB, and the top right of each image shows a magnified region outlined by the blue box in the ground truth image.

C. In vivo Experiments

We consider a challenging dataset with real images containing large motions to test the methods further. The *in vivo* Ultrafast Doppler sequences depict human patients who have undergone brain surgery at the neurosurgery department of the University Hospital of Tours (UHT). The data are part of the ELASTOGLI cohort [44], which was registered on clinical trial.gov (ntC03970499) and approved by the French Ethics Committee (CPP Ile-de-France IDF1-2018-ND42-cat.2).

Data acquisition was performed after the *craniotomy* stage in which the dura mater was also removed. Two regions are of interest in the images: a healthy brain area with mainly large cerebral arteries and a region with glioma having complex vascular structures and minimal flow in small vessels (see Figure. 3).

1) Data Acquisition: The acquisitions were made using the AixplorerTM (Supersonic Imagine) ultrasound scanner equipped with an SL10-2 probe (192 elements). A sequence of 1000 frames was uploaded on the scanner using the research package (SonicLab V12), with $[-5^{\mathrm{o}}, 0^{\mathrm{o}}, +5^{\mathrm{o}}]$ compounded angles, a pulse repetition frequency (PRF) of 3KHz, a frame rate of 1KHz, and an imaging depth of $[1\mathrm{mm} - 40\mathrm{mm}]$. The acquired sequences were downloaded on a hard disk for offline beamforming and signal processing.

Two sequences of raw in-phase and quadrature (IQ) images of size $260 \times 192 \times 1000$ pixels were recorded. The first sequence (*Invivo1*) spans a one-second acquisition period, while the second (*Invivo2*) lasts two seconds. Due to its longer duration, *Invivo2* captures significantly more pronounced respiratory motions and blood pulsatility compared to *Invivo1*. Both sequences depict a glioma. However, *Invivo1*, displays a larger, poorly vascularized central area and some necrotic tissues. The central frame of each sequence is shown in the first column of Figure 3.

2) Results: No ground truth was available for the *in vivo* sequences. Therefore, we use the contrast ratio (CR), a measure of contrast between the background and blood regions [9], [10]. A larger CR value indicates a better separation of blood and tissues. In practice, CR values are computed for rectangular patches of size 13×12 pixels covering the whole image and then averaged. For a pair of background and foreground patches, the CR is obtained as follows:

$$\mathrm{CR}_{[\mathrm{dB}]} = 20 \log_{10}(m_f / m_b) \tag{16}$$

where m_f and m_b are the mean intensity values of the foreground and background patches, respectively. As the same background patch must be selected for all methods, we use the one that maximizes the CR for the SVD method.

Fig. 4 shows the boxplots of all the patchwise CR values obtained for both sequences, and Table III shows the median CR values for each sequence separately. One can see that the proposed method results in overall better contrasts, producing higher median values in all cases. The Piepenbrock and Demon methods, which use a motion compensation step, provide the second-best results. These results confirm that motion estimation is also beneficial for real data. Note that we focus on the Tikhonov regularization for the proposed method as it has shown equal or improved performance compared to the TV ones in our previous experiments. Moreover, the LK+RPCA and DeepfUS methods performed poorly compared to the other methods for all tested parameter settings; the results are therefore not shown for lack of space. Note that

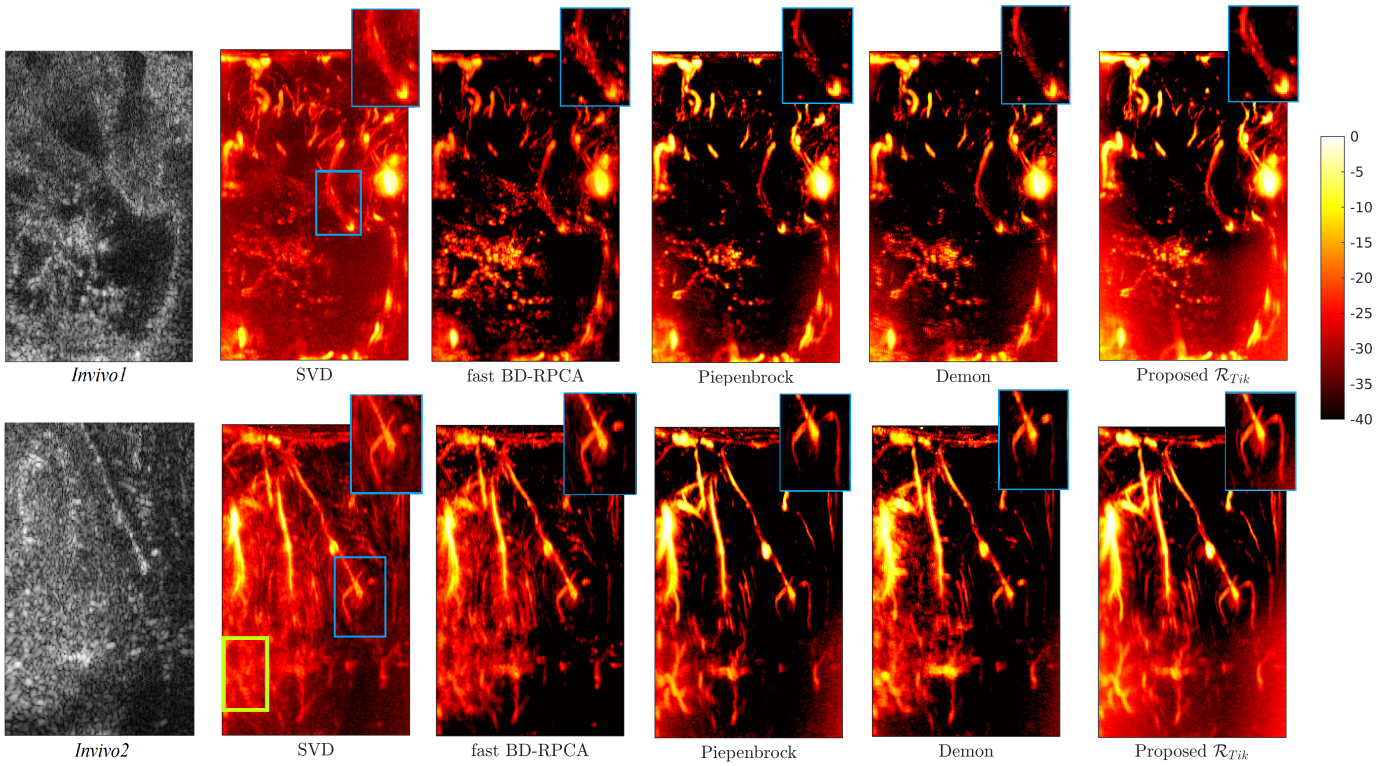

Fig. 3. Central frames (1st column, intensities are in the range $[0, 1]$) and Power Doppler images obtained for the *Invivo1* (1st row) and *Invivo2* (2st row) sequences using: SVD, fBDRPCA, Piepenbrock, Demon, and the proposed method. Magnified examples of enhanced structures can be seen in the top-right parts of the images.

DeepfUS uses a pre-trained network with fixed image size and sequence length. Therefore, we used overlapping patches to process the *in vivo* sequences. A temporal sub-sampling (by a factor of 8) was also necessary. Various subsampling methods were tested without succeeding in recovering the blood flow, suggesting that a retraining of the network on similar *in vivo* sequences may be necessary to obtain meaningful results.

Fig. 3 shows the Power Doppler images obtained by the remaining methods. Overall, the visual results are consistent with the CR values in Table III; for example, SVD resulted clearly in the lowest contrast. The proposed method provided sharp edges and a brighter blood flow than others in several areas of the Power Doppler image. The magnified parts (blue box) illustrate how the proposed method reveals certain vascular structures more efficiently than the others. A green box highlights another example in the SVD result for *Invivo2*, showing the benefits of the proposed method in deeper regions. While methods not using any motion compensation failed to recover these structures, Piepenbrock and Demon recovered a blurred version. In contrast, the proposed method provided enhanced results for these structures. This is even more noticeable for *Invivo2*, where the motions are amplified. It is worth noting that enhancing the CR comes at the cost of losing a few smaller vessels. This suggests a trade-off between contrast and small vessel preservation, which is a direction of improvement for the proposed method. However, while methods such as SVD suffer from large amounts of clutter when revealing small structures, we have observed that the trade-off required by the proposed method to do so caused considerably less significant SNR drops.

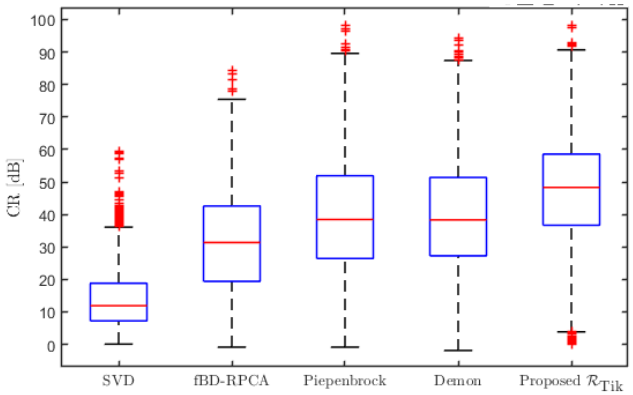

Fig. 4. Boxplots of the CRs for the *in vivo* sequences. The red horizontal lines indicate the median values, and the red markers represent outliers.

D. Execution Times

Tables III-IV present the average execution times of the MATLAB codes run on a computer with an Intel(R) Xeon(R) W-2235 CPU processor @3.8MHz using 251GB of RAM, running Linux v5.4.0. One can see that SVD is the fastest algorithm but also the least accurate for all experiments. The execution times of the proposed approach are competitive with the rest of the algorithms; in particular, the Tikhonov regularization was faster than fBD-RPCA and Demon for

TABLE III
$\mathrm{CR}_{[\mathrm{DB}]}$ AND AVERAGE EXECUTION TIMES FOR THE *in vivo* DATA.

Method	SVD [5]	fBD-RPCA [10]	Piepenbrock [25]	Demon [39]	DeepfUS [23]	Proposed $\mathcal{R}_{\mathrm{Tik}}$
Invivo1	10.45	31.93	36.68	35.45	2.87	**47.51**
Invivo2	13.63	30.56	41.10	41.58	2.21	**49.35**
Times (s)	**1.71**	13.23	52.80	310.44	42	162.10

TABLE IV
AVERAGE EXECUTION TIMES (S) FOR THE SYNTHETIC EXPERIMENTS.

Method	Synthetic T	Synthetic B
SVD [5]	**0.27**	**0.28**
fBD-RPCA [10]	76.3	88.5
LK+RPCA [25]	19.4	22.0
Piepenbrock [25]	21.1	19.1
Demon [39]	81.7	81.5
DeepfUS [23]	2.32	2.29
Proposed $\mathcal{R}_{\mathrm{Tik}}$	28.8	28.2
Proposed $\mathcal{R}_{\mathrm{TV}-\ell_1}$	101.9	73.2
Proposed $\mathcal{R}_{\mathrm{TV}-\ell_2}$	110.5	74.7

the synthetic experiments. The TV regularizations required longer execution times (due to the additional iterations of Algorithm 2). Still, their execution times were comparable to fBD-RPCA and Demon, which performed, in some cases, relatively better than other methods proposed for comparison. Generally, the *in vivo* experiments resulted in longer execution times due to the size of the sequences (1000 frames). The proposed method required longer execution times[8] than most methods, but it was faster than the Demon algorithm. Finally, the above results suggest that the Tikhonov regularization is a good choice for the considered datasets, with competitive execution times and improved performance.

V. CONCLUSIONS AND PERSPECTIVES

This paper introduced a novel method for computing blood flow in moving UUI sequences. The proposed method jointly estimates blood, tissues, and their motions, using an inverse problem formulation considering each component's specificity and exploring various regularizations. A carefully designed convex minimization problem was solved using a proximal algorithm. A comparison with recent techniques showed a significant improvement in blood flow estimation quality in the presence of moving tissues for simulated data. Moreover, the study of different smoothness constraints demonstrated that the Tikhonov regularization was the best in accuracy and speed. *In vivo* investigations suggested a trade-off between preserving small vascular structures and good image contrast. Studying this trade-off with more clinical data will be the focus of future work. The degradation caused by the PSF could also be studied, for example, by developing a joint blind deconvolution approach. Implementation on GPUs would substantially decrease the execution time, and differential programming techniques would enable automatic hyperparameter tuning, alleviating a practical downside of current state-of-the-art methods.

[8]These execution times can be reduced by limiting the number of iterations where the motions are updated (1 update every 10 iterations, for example) while preserving all convergence guarantees, which is a nice property of the proposed proximal approach.

ACKNOWLEDGEMENT

The authors thank Dr J.P. Remeniéras and G. Lacoin from the INSERM IBrain Group at the University of Tours for providing the *in vivo* data.

APPENDIX A
LIPSCHITZ CONSTANTS

It follows from (6), that the Lipschitz constant[9] of $\boldsymbol{B} \mapsto h(\boldsymbol{B}+\boldsymbol{T},\boldsymbol{d})$ is

$$L_d = 2 \max_{t\in\{1,\dots,N_t\}} \|\boldsymbol{I}\|_{\mathrm{S}}^2 = 2, \tag{17}$$

where $\|\cdot\|_{\mathrm{S}}$ denotes the spectral norm. By symmetry, the same Lipschitz constant is obtained for $\boldsymbol{T} \mapsto h(\boldsymbol{B}+\boldsymbol{T},\boldsymbol{d})$. For the function $\boldsymbol{d} \mapsto h(\boldsymbol{B}+\boldsymbol{T},\boldsymbol{d})$, we have

$$L_{B,T} = 2 \max_{t\in\{1,\dots,N_t\},\boldsymbol{y}\in\mathcal{Y}} \|\boldsymbol{\nabla} S(\boldsymbol{y},t)\|^2, \tag{18}$$

which is computed only once. Note that the proposed model allows us to obtain constant step sizes, simplifying the algorithm implementation.

APPENDIX B
PROXIMITY OPERATORS OF THE SPARSITY AND LOW-RANK CONSTRAINTS

The proximity operator of the $\ell_{1,2}$-norm is the row-shrinkage operator.[10] The corresponding proximal step at iteration k of Algorithm 1 is computed as follows

$$\mathrm{prox}_{(\sigma_B^k)^{-1},\lambda\|\cdot\|_{1,2}}(\tilde{\boldsymbol{B}}^k) = \left[\left(1-\frac{\sigma_B^k\lambda}{\max(\|\tilde{\boldsymbol{B}}_i^k\|_2,\sigma_B^k\lambda)}\right)\tilde{\boldsymbol{B}}_i^k\right]_{1\le i\le N_xN_z}, \tag{19}$$

where $\tilde{\boldsymbol{B}}_i^k$ is the i-th row of $\tilde{\boldsymbol{B}}^k$.

The proximity operator of the nuclear norm $\|\cdot\|_*$ performs a soft-thresholding on the singular values of the Casorati matrix $\tilde{\boldsymbol{T}}^k$. The corresponding step in the proposed algorithm is computed as

$$\mathrm{prox}_{(\sigma_T^k)^{-1},\mu\|.\|_*}(\tilde{\boldsymbol{T}}^k) = \boldsymbol{U}\mathrm{prox}_{(\sigma_T^k)^{-1},\mu\|.\|_1}(\boldsymbol{\Sigma})\boldsymbol{V}^T, \tag{20}$$

where $\tilde{\boldsymbol{T}}^k = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ is the singular value decomposition (SVD) of $\tilde{\boldsymbol{T}}^k$ and

$$\mathrm{prox}_{(\sigma_T^k)^{-1},\mu\|.\|_1}(\boldsymbol{\Sigma}) = \mathrm{sign}(\boldsymbol{\Sigma}) \odot \max(|\boldsymbol{\Sigma}| - \sigma_T^k\mu, 0), \tag{21}$$

[9]L is called the Lipschitz constant of a function $(x,y) \mapsto h(x,y)$ in the variable y over a set $\mathcal{S}$ with respect to a norm $\|\cdot\|$ if $(\forall (y_1,y_2) \in \mathcal{S})\,, \|f(x,y_1)-f(x,y_2)\| \le L\|y_1-y_2\|$. L is a measure of the maximum rate of change of a function.

[10]See http://proximity-operator.net

where sign, max, and their Hadamard product $\odot$ are element-wise operations.

APPENDIX C
MOTION REGULARIZATION WITH $\mathcal{R}_{\text{TIK}}$

The Tikhonov regularization defined in (9) ($\mathcal{R}_{\text{Tik}}$) is a convex and smooth function. Therefore, this term belongs to the smooth part of the cost function (12) and can be integrated directly into the gradient step with

$$\nabla_{\boldsymbol{d}_t}\mathcal{R}_{\text{Tik}}(\boldsymbol{d}) = 2\left(\begin{bmatrix}\boldsymbol{G}^\top\boldsymbol{G}d_x(t)\\ \boldsymbol{G}^\top\boldsymbol{G}d_z(t)\end{bmatrix}\right)_{1\leq t\leq N_t}. \tag{22}$$

In this case, the proximal update of step 3) in Algorithm 1 reduces to the proximity operator of the box constraint, *i.e.,* the projection onto the constraint set $[d_{\min}, d_{\max}]^{2N_xN_z}$ as follows

$$\boldsymbol{d}_t^{k+1} = P_{[d_{\min},d_{\max}]^{2N_xN_z}}(\boldsymbol{d}_t) \tag{23}$$

$$= \Big[\max\{d_{\min}, \min(\boldsymbol{d}_{t,i}, d_{\max})\}\Big]_{\substack{1\leq t\leq N_t,\\ 1\leq i\leq 2N_xN_z}} \tag{24}$$

REFERENCES

[1] M. Imbault, D. Chauvet, J.-L. Gennisson, L. Capelle, and M. Tanter, "Intraoperative Functional Ultrasound Imaging of Human Brain Activity," *Scientific Reports*, vol. 7, no. 1, p. 7304, Aug. 2017.

[2] C. Barthelemy, E. Koury, S. Beuve, I. Zemmoura, J. Gennisson, A. Bassarab, D. Kouame, and J. Remenieras, "Development of ultrasensitive doppler imaging method for the surgical management of open-brain tumors," in *Proc. IEEE Ultrason. Symp. (IUS)*, IEEE, Oct. 2019.

[3] S. Soloukey, A. J. P. E. Vincent, D. D. Satoer, F. Mastik, M. Smits, C. M. F. Dirven, C. Strydis, J. G. Bosch, A. F. W. Van Der Steen, C. I. De Zeeuw, S. K. E. Koekkoek, and P. Kruizinga, "Functional Ultrasound (fUS) During Awake Brain Surgery: The Clinical Potential of Intra-Operative Functional and Vascular Brain Mapping," *Frontiers in Neuroscience*, vol. 13, p. 1384, Jan. 2020.

[4] E. Mace, G. Montaldo, B. Osmanski, I. Cohen, M. Fink, and M. Tanter, "Functional ultrasound imaging of the brain: Theory and basic principles," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 60, no. 3, pp. 492–506, 2013.

[5] C. Demené, T. Deffieux, M. Pernot, B. F. Osmanski, V. Biran, J. L. Gennisson, L. A. Sieu, A. Bergel, S. Franqui, J. M. Correas, I. Cohen, O. Baud, and M. Tanter, "Spatiotemporal clutter filtering of ultrafast ultrasound data highly increases doppler and fultrasound sensitivity," *IEEE Trans. Med. Imag.*, vol. 34, no. 11, pp. 2271–2285, Nov. 2015.

[6] J. Baranger, B. Arnal, F. Perren, O. Baud, M. Tanter, and C. Demené, "Adaptive spatiotemporal svd clutter filtering for ultrafast doppler imaging using similarity of spatial singular vectors," *IEEE Trans. Med. Imag.*, vol. 37, no. 7, pp. 1574–1586, 2018.

[7] M. Ashikuzzaman, C. Belasso, M. G. Kibria, A. Bergdahl, C. J. Gauthier, and H. Rivaz, "Low rank and sparse decomposition of ultrasound color flow images for suppressing clutter in real-time," *IEEE Trans. Med. Imag.*, vol. 39, no. 4, pp. 1073–1084, 2020.

[8] H. Shen, C. Barthelemy, E. Khoury, Y. Zemmoura, J. Remenieras, A. Basarab, and D. Kouamé, "High-resolution and high-sensitivity blood flow estimation using optimization approaches with application to vascularization imaging," in *Proc. IEEE Int. Ultrason. Symp. (IUS)*, 2019, pp. 467–470.

[9] D. H. Pham, A. Basarab, I. Zemmoura, J.-P. Remenieras, and D. Kouame, "Joint blind deconvolution and robust principal component analysis for blood flow estimation in medical ultrasound imaging," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 68, no. 4, pp. 969–978, Apr. 2021.

[10] D. H. Pham, A. Basarab, J.-P. Remenieras, P. Rodríguez, and D. Kouamé, "Fast high resolution blood flow estimation and clutter rejection via an alternating optimization problem," in *Proc. IEEE 18th Int. Symp. Biomed. Imag. (ISBI)*, 2021, pp. 1907–1910.

[11] H. Hasegawa and C. de Korte, "Special issue on ultrafast ultrasound imaging and its applications," *Appl. Sci.*, vol. 8, no. 7, p. 1110, Jul. 2018.

[12] J. Bercoff, "Ultrafast ultrasound imaging," in *Ultrasound Imaging - Medical Applications*, InTech, Aug. 2011.

[13] M. Tanter and M. Fink, "Ultrafast imaging in biomedical ultrasound," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, pp. 102–119, 2014.

[14] L. A. Ledoux, P. J. Brands, and A. P. Hoeks, "Reduction of the clutter component in doppler ultrasound signals based on singular value decomposition: A simulation study," *Ultraso. Imag.*, vol. 19, no. 1, pp. 1–18, Jan. 1997.

[15] S. Bjaerum, H. Torp, and K. Kristoffersen, "Clutter filter design for ultrasound color flow imaging," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 49, no. 2, pp. 204–216, 2002.

[16] L. Lovstakken, S. Bjaerum, K. Kristoffersen, R. Haaverstad, and H. Torp, "Real-time adaptive clutter rejection filtering in color flow imaging using power method iterations," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 53, no. 9, pp. 1597–1608, 2006.

[17] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," *Proc. Neural Inf. Process. Syst*, pp. 1–9, 2009.

[18] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, pp. 1–37, May 2011.

[19] M. Grosser, M. Möddel, and T. Knopp, "Using low-rank tensors for the recovery of MPI system matrices," *IEEE Trans. Comput. Imaging*, vol. 6, pp. 1389–1402, 2020.

[20] C. G. Tsinos, A. A. Rontogiannis, and K. Berberidis, "Distributed blind hyperspectral unmixing via joint sparsity and low-rank constrained non-negative matrix factorization," *IEEE Trans. Comput. Imaging*, vol. 3, no. 2, pp. 160–174, 2017.

[21] M. Bayat and M. Fatemi, "Concurrent clutter and noise suppression via low-rank plus sparse optimization for non-contrast ultrasound flow doppler processing in microvasculature," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Calgary, Canada, Apr. 2018.

[22] S. G. Sathyanarayna, S. T. Acton, and J. A. Hossack, "Suppression of clutter by rank adaptive reweighted sparse coding," in *Proc. IEEE Ultrason. Symp. IUS*, Washington DC, USA, Sep. 2017.

[23] T. Di Ianni and R. D. Airan, "Deep-fus: A deep learning platform for functional ultrasound imaging of the brain using sparse data," *IEEE Trans. Med. Imaging*, vol. 41, no. 7, pp. 1813–1825, 2022.

[24] M. Blons, T. Deffieux, B.-F. Osmanski, M. Tanter, and B. Berthon, "Perceptflow: Real-time ultrafast doppler image enhancement using deep convolutional neural network and perceptual loss," *Ultrasound in Medicine & Biology*, vol. 49, no. 1, pp. 225–236, 2023.

[25] M. Piepenbrock, S. Dencks, and G. Schmitz, "Tissue motion estimation of contrast enhanced ultrasound images with a stable principal component pursuit," in *Proc. IEEE 18th Int. Symp. Biomed. Imag. (ISBI)*, 2021, pp. 1642–1645.

[26] S. Harput, K. Christensen-Jeffries, J. Brown, Y. Li, K. J. Williams, A. H. Davies, R. J. Eckersley, C. Dunsby, and M.-X. Tang, "Two-stage motion correction for super-resolution ultrasound imaging in human lower limb," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 65, no. 5, pp. 803–814, May 2018.

[27] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, no. 1–3, pp. 185–203, Aug. 1981.

[28] N. Ouzir, A. Basarab, O. Lairez, and J.-Y. Tourneret, "Robust optical flow estimation in cardiac ultrasound images using a sparse representation," *IEEE Trans. Med. Imag.*, vol. 38, no. 3, pp. 741–752, 2019.

[29] A. Goujon, S. Neumayer, P. Bohra, S. Ducotterd, and M. Unser, "A neural-network-based convex regularizer for inverse problems," *IEEE Trans. Comput. Imaging*, vol. 9, pp. 781–795, 2023.

[30] V. Nikitin, V. De Andrade, A. Slyamov, B. J. Gould, Y. Zhang, V. Sampathkumar, N. Kasthuri, D. Gürsoy, and F. De Carlo, "Distributed optimization for nonrigid nano-tomography," *IEEE Trans. Comput. Imaging*, vol. 7, pp. 272–287, 2021.

[31] H. H. Bauschke and P. L. Combettes, "Convex analysis and monotone operator theory in hilbert spaces," in *2nd ed., corrected printing*. New York: Springer International Publishing, 2019, pp. C1–C4.

[32] C. Chaux, M. el gheche, J. Farah, J.-C. Pesquet, and B. Pesquet, "A parallel proximal splitting method for disparity estimation from multicomponent images under illumination variation," *J. Math. Imaging Vis.*, vol. 47, pp. 167–178, Nov. 2013.

[33] F. Abboud, E. Chouzenoux, J.-C. Pesquet, J.-H. Chenot, and L. Laborelli, "Dual block-coordinate forward-backward algorithm with application to deconvolution and deinterlacing of video sequences," *J. Math. Imaging Vis.*, vol. 59, pp. 415–431, 2016.

[34] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Mathematical Programming*, vol. 146, no. 1-2, pp. 459–494, Jul. 2013.

[35] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, "A block coordinate variable metric forward-backward algorithm," *J. Glob. Optim.*, pp. 1–29, Feb. 2016.

[36] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, Eds. New York, NY: Springer New York, 2011, pp. 185–212.

[37] S. Shalev-Shwartz and T. Zhang, "Stochastic dual coordinate ascent methods for regularized loss," *J. Mach. Learn. Res.*, vol. 14, no. 1, p. 33, Feb. 2013.

[38] A. Chambolle and C. Dossal, "On the convergence of the iterates of the "fast iterative shrinkage/thresholding algorithm"," *J. Optim. Theory Appl.*, vol. 166, no. 3, pp. 968–982, May 2015.

[39] V. Pustovalov, D.-H. Pham, J.-P. Remenieras, and D. Kouamé, "Motion compensation for the estimation of high-resolution blood flow in ultrafast ultrasound imaging," in *Medical Imaging 2022: Ultrasonic Imaging and Tomography*, SPIE, Apr. 2022.

[40] J. Meunier and M. Bertrand, "Echographic image mean gray level changes with tissue dynamics: A system-based model study," *IEEE Trans. Biomed. Eng.*, vol. 42, no. 4, pp. 403–410, 1995.

[41] ——, "Ultrasonic texture motion analysis: Theory and simulation," *IEEE Trans. Med. Imaging*, vol. 14, no. 2, pp. 293–300, 1995.

[42] A. Marion and D. Vray, "Toward a real-time simulation of ultrasound image sequences based on a 3-d set of moving scatterers," *Trans. Ultrason., Ferroelect., Freq. Control*, vol. 56, no. 10, pp. 2167–2179, 2009.

[43] J. Udesen, F. Gran, K. L. Hansen, J. A. Jensen, C. Thomsen, and M. B. Nielsen, "High frame-rate blood vector velocity imaging using plane waves: Simulations and preliminary experiments," *Trans. Ultrason., Ferroelect., Freq. Control*, vol. 55, no. 8, pp. 1729–1743, 2008.

[44] L. Guillaume, Z. Ilyess, G. Jean-Luc, K. Denis, and R. Jean-Pierre, "Multi-layered adaptive neoangiogenesis Intra-Operative quantification (MANIOQ)," *Journal of Cerebral Blood Flow & Metabolism*, 2023.