

RESEARCH ARTICLE

Software for non-parametric image registration of 2-photon imaging data

Philipp Flotho^{1,2,3*}  | Shinobu Nomura⁴  | Bernd Kuhn⁴  | Daniel J. Strauss^{1,3} 

¹Systems Neuroscience and Neurotechnology Unit, Neurocenter, Faculty of Medicine, Saarland University & School of Engineering, htw saar, Germany

²Summer Program, Japan Society for the Promotion of Science (JSPS), Tokyo

³Center for Digital Neurotechnologies Saar (CDNS), Homburg, Germany

⁴Optical Neuroimaging Unit, Okinawa Institute of Science and Technology Graduate University, Onna, Okinawa

***Correspondence**

Philipp Flotho, Systems Neuroscience and Neurotechnology Unit, Neurocenter, Faculty of Medicine, Saarland University and School of Engineering, htw saar, Germany.

Email: philipp.flotho@uni-saarland.de

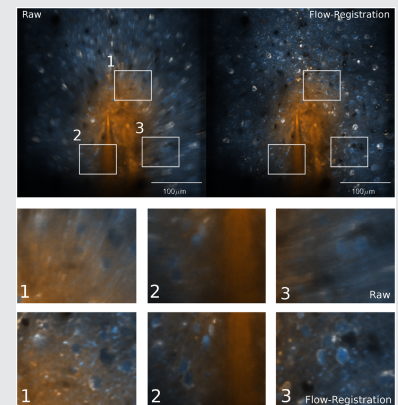
Abstract

Functional 2-photon microscopy is a key technology for imaging neuronal activity. The recorded image sequences, however, can contain non-rigid movement artifacts which requires high-accuracy movement correction. Variational *optical flow* (OF) estimation is a group of methods for motion analysis with established performance in many computer vision areas. However, it has yet to be adapted to the statistics of 2-photon neuroimaging data.

In this work, we present the motion compensation method *Flow-Registration* that outperforms previous alignment tools and allows to align and reconstruct even low *signal-to-noise ratio* 2-photon imaging data and is able to compensate high-divergence displacements during local drug injections. The method is based on statistics of such data and integrates previous advances in variational OF estimation. Our method is available as an easy-to-use ImageJ/FIJI plugin as well as a MATLAB toolbox with modular, object oriented file IO, native multi-channel support and compatibility with existing 2-photon imaging suites.

KEYWORDS

confocal microscopy, image registration, ImageJ/FIJI plugin, MATLAB toolbox, movement correction, optical flow, optical imaging, two-photon microscopy



Abbreviations: EPE, endpoint error; MSE, mean squared error; OF, optical flow; PSNR, peak signal to noise ratio; ROI, region of interest; SNR, signal to noise ratio.

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2022 The Authors. *Journal of Biophotonics* published by Wiley-VCH GmbH.

1 | INTRODUCTION

2-photon microscopy in combination with synthetic or genetically encoded indicators allows to image a wide range of different aspects of neuronal activity with

cellular or even sub-cellular resolution in anesthetized as well as behaving animals.^[1,2] Importantly, small signal changes might carry crucial information. However, the imaging data can be afflicted with different types of noise and artifacts. On the one hand, due to the low number of generated photons with 2-photon excitation, the shot noise is typically significant. On the other hand, movement noise can be introduced during acquisition. Motion artifacts can be caused by heart beat, breathing, as well as motor behavior in awake animals. Also, some experimental paradigms inherently result in large, non-rigid and non-elastic deformations, for example, local drug injections. While many established tools exist for the correction of small vibrations and rigid drifts, the compensation of large and/or non-uniform motion is still a challenge.

Furthermore, for high accuracy alignment, subpixel registration is necessary: Small displacements can be approximated by a linearized motion assumption.^[3] This means even small residual motion can induce large artifacts around image edges due to a proportional relation of spatial image gradient and motion magnitude with respect to induced brightness changes.

While *optical flow* (OF) based image registration methods were used for 2-photon imaging data before,^[4] they do not incorporate the advances in OF estimation developed in recent years in the context of many computer vision areas.

As a consequence, they perform poorly, when compared with state-of-the-art image registration methods for 2-photon imaging and are generally regarded as too prone to noise for this application.^[5]

In our previous work,^[6] we developed a variational motion compensation strategy for 1D linescans (e.g.^[7]) and could already demonstrate that metrics from variational OF estimation can be robust enough for such low SNR applications.

In the context of image registration, images are aligned with different metrics and constraints on transformations. To enable reproducible research, the image registration toolbox *elastix* is built around an environment that enables different setting configurations^[8] which can be downloaded from a model zoo. *Advanced normalization tools* (ANTs) is another toolbox for traditional image registration which allows to configure multiple different metrics and dataterms.^[9] ANTs has proven high performance for brain image registration and labeling of brain scans.^[10,11]

There has been a great deal of work on OF techniques in the past decades with the goal of improving accuracy, model invariants as well as robustness and computing speed. The recent advancements in OF estimation tackle the problem of large displacements with discontinuous motion fields and motion layers. Chen et al.^[12] still lead (as of March 2022) the Middlebury optical flow benchmark^[13]

with respect to average *endpoint error* (EPE) and average angular error. They use similarity transformations for a segmented flow field as initialization of large motion. In a second step, the variational method of Sun et al.^[14] is used for subpixel refinement.

In this work, we propose a novel, OF-based image registration approach for the motion compensation of 2-Photon neuroimaging data. We build on the well studied framework for variational OF estimation^[3,14,15] and adapt this framework to 2-photon imaging data by techniques which have recently been developed in visual computing. We demonstrate the performance on challenging 2-photon imaging data and can report state-of-the-art results in terms of registration quality, competing computation speed and easy accessibility. Our method is available as an easy-to-use MATLAB toolbox as well as an ImageJ/FIJI plugin.

2 | MATERIALS AND METHODS

In this section, we will first briefly describe the acquisition methods for the data that is used in the benchmark dataset and then present the theory as well as implementation details of the proposed motion compensation method Flow-Registration.

2.1 | Animals

Experiments were approved by the OIST Institutional Animal Care and Use Committee, and performed in and AAALAC International accredited facility. Animals were maintained in a 12 h/12 h light/dark cycle at 22°C, with food and water available ad libitum.

2.2 | Recombinant viruses

The adeno-associated virus (AAV) encoding the GAKdYmut PKA activity sensor^[16] was custom made and produced by the vector core facility of Pennsylvania University (AAV2/1-hSyn-GAKdYmut-hGH, titer: 4×10^{12} gc/mL), and mixed with AAV2/1-hSyn-TurboRFP-WPRE encoding the red fluorophore RFP (titer: 4×10^{13} gc/mL, same supplier) or with AAV1.Syn.NES-jRGECO1a.WPRE.SV40 encoding the red calcium indicator jRGECO (titer: 3×10^{13} gc/mL, same supplier) at a ratio 1: 3.

2.3 | Expression of GAKdYmut and TurboRFP in cortex

Viral transfer of the indicator gene into cortical neurons of the mouse was performed as described before.^[17] C57/

BL6 mice (2-month-old) were anesthetized with a mixture of medetomidine (0.3 mg/kg), midazolam (4 mg/kg), and butorphanol (5 mg/kg). After performing a craniotomy over somatosensory cortex (AP -1.5 mm, ML 1.7 mm, DV -0.6-0.7 mm from bregma), 70–140 nL of a 1:3 mixture of AAV2/1-hSyn-GAkdYmut-hGH and AAV2/1-hSyn-TurboRFP-WPRE or AAV1.Syn.NES-jRGECO1a.WPRE.SV40 was injected in layer V at a rate of 10 nL/min. A chronic cranial window with a silicon access port (5 mm glass coverslip) was mounted as described earlier.^[18,19] At the end of the surgery, mice received atipamezole (0.3 mg/kg) for recovery from anesthesia, and buprenorphine (0.1 mg/kg) for pain relieve. Five to eight weeks after the AAV injection, mice were head-fixed for imaging experiments performed under anesthesia with 1% isoflurane or awake.

2.4 | In vivo imaging in cortex

A combined wide-field/two-photon microscope (MOM, Sutter Instruments) with a femtosecond-pulsed Ti:sapphire laser (Vision II, Coherent) was used. To increase the point spread function of excitation the back aperture of the 25× water immersion objective (Olympus) was underfilled (spatial resolution $1\mu\text{m} \times 1\mu\text{m} \times 4\mu\text{m}$). The collar of the objective was adjusted to correct for the window glass thickness (170 μm). Simultaneous excitation of GAkdYmut (GFP-based single fluorophore sensor^[16]) and TurboRFP was performed at 950 nm with a typical power of 5–11 mW. Fluorescence was detected in two channels by GaAsP photomultipliers (Hamamatsu) in spectral windows 490–550 nm (GAkdYmut) and 600–700 nm (TurboRFP), separated by a 560 nm dichroic mirror (all Semrock). The microscope was controlled by a commercial software (MScan, Sutter Instruments). Sampling rate was 30.9 frame/s with 512×512 pixel, corresponding to a field of view of $375 \mu\text{m} \times 375 \mu\text{m}$. Saline (0.9% NaCl) with Alexa592 (1 μm) or with additional drug (propranolol 10 mM) was injected under anesthesia. Pressure injection was performed through the silicon access port using a glass pipette beveled to a diameter of 5–10 μm opening.

2.5 | Motion compensation method

2.5.1 | Statistics in 2-photon microscopy

The motion statistics for 2-photon imaging differ from the motion we encounter in natural images in several ways: Due to the illumination strategy, there is only a single imaging plane and thus no different motion layers.

Also, the imaged object is usually soft, biological tissue and, as a consequence, we expect smooth motion fields without discontinuities. We expect the image to contain a fixed *region of interest* (ROI) with small displacements between frames and, potentially, large drift over time which can result in large displacements with respect to the reference. Due to the nature of the scanning method, horizontal displacements may occur. Also, the images are usually not represented in perceptual color spaces, restricting the use of classical photometric invariances, such as in,^[20] but, often, there exist multiple signal and or structural channels. Depending on the indicators used, the different channels can contain disjoint structures and different SNR characteristics. Due to technical limitations of current generation imaging, high speed imaging can often only be realized on narrow field of views (FOV).

Our image registration method builds on those observations: Due to the absence of different motion layers and temporally coherent recordings, we do neither need an initialization strategy with a more robust method such as in^[12] nor a regularizer that preserves discontinuities (e.g.^[14,20]). Also, the assumptions of elastic regularizers which penalize the divergence of the OF field, for example, compare,^[21] are violated by recordings during drug injection, see Figure 1F, where the displacement field has a high divergence after injection. Additionally, we generalize our previous linescan alignment^[6] to recordings with narrow FOV and propose a non-uniform warping strategy where we perform more warping steps along the larger image dimension during optimization.

2.5.2 | Optical flow method

Let $f: (0, W) \times (0, H) \rightarrow \mathbb{R}^c$ be a moving and $f_{\text{ref}}: (0, W) \times (0, H) \rightarrow \mathbb{R}^c$ be a fixed image with width W , height H and c channels. We then depict the displacement field that points from the reference to the moving image as $w: (0, W) \times (0, H) \rightarrow \mathbb{R}^2$. In the following, we use a compact notation and omit the function arguments if not needed. We use coordinates $x = (x_1, x_2)^\top$ and depict the components of the displacement field with $w(x) = (u(x), v(x))^\top$. The goal of variational OF is to estimate w such that $f_{\text{ref}}(x)$ depicts the same scene point as $f(x+w)$ by minimizing an energy functional that consists of a dataterm $D(w)$, a smoothness term $S(w)$ and usually a smoothness weight α . The dataterm penalizes deviations from constancy assumptions and the smoothness term imposes smoothness on the results in image areas that do not contain enough information (aperture problem^[20]). The energy functional for variational OF usually has the form

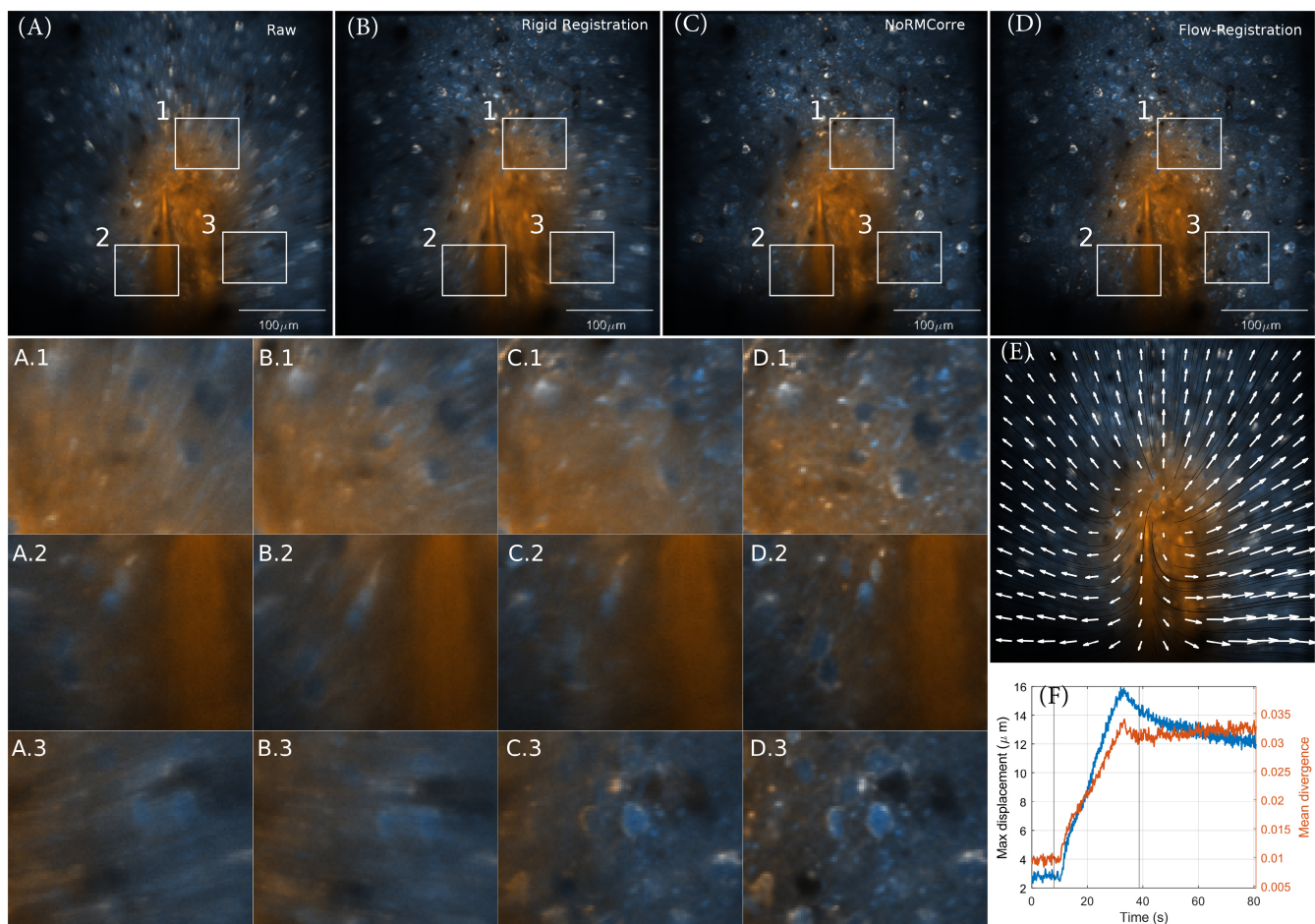


FIGURE 1 Comparison of registration performance on a very challenging two-channel 2-photon recording during drug injection in vivo. The challenges of the sequence are a very low signal-to-noise ratio together with brightness changes (orange) and non-elastic deformations due to the injection. Average of (A) raw images, (B) after rigid registration, (C) after registration with NoRMCorre, and (D) after Flow-Registration. The tissue expands from the injection point (E) resulting in large displacements as well as in a high divergence (F) in the displacement field. Flow-Registration can recover fine structures with much more detail, allowing region-of-interest selection along fine structures, while the blur and double images in (A-C) indicate residual motion. This is well visible for the fine structures and grains in D.1 which are smeared by motion blur in A.1-C.1 and become more difficult to observe. The channels in the false color representation have all been normalized with respect to the min and max intensity values of the averaged raw recording. Neurons express the PKA sensor GAKdYmut (green channel, mapped to blue) and the red chromophore TurboRFP (red channel, mapped to orange); the injection contained the fluorescent dye Alexa 594 (red channel, orange). See Appendix for additional red-green and checkerboard visualizations of the dataset as well as a description of the color mapping

$$E(w) = \int D(w) + \alpha S(w) dx \quad (1)$$

We model our observations as a variational OF method with gradient constancy, robust, separate channel penalization of the dataterm and first-order regularizer that we set to homogeneous diffusion. We normalize the dataterm according to Zimmer et al.^[20] The regularizer and dataterm are penalized with a generalized Charbonnier penalty function which for $\epsilon > 0$ is given by $\Psi_a(x) = (x^2 + \epsilon^2)^a$, for $a > 0$.^[14] We use a_{smooth} to depict the parameter for the smoothness and a_{data} to depict the parameter for the dataterm. Gradient constancy assumes that the gradients of the imaged structures do not change with respect to

the reference. For a moving image f and a fixed image f^{ref} , a data term that quadratically penalizes constancy of the image gradients is given by

$$d(w, x) = \left(f_{x_1}(x+w) - f_{x_1}^{\text{ref}}(x) \right)^2 + \left(f_{x_2}(x+w) - f_{x_2}^{\text{ref}}(x) \right)^2 \quad (2)$$

In the variational framework, we can minimize the joint energy of the structural and the functional channels. This has the advantage that, in theory, we get a better SNR, if the same structures are visible in both channels and, additionally, considering the aperture problem, we

TABLE 1 Average endpoint errors of the estimated displacements (the lower the better). The value indicates the average distance in pixels of the estimated displacements with respect to the ground truth. The best performance per column is put in bold and the best performance per dataset is underlined. Channel ch1 simulates the functional channel with positive multiplicative (drug injection) and ch2 the structural with negative multiplicative changes (z-shift). 35 dB is comparable to the 6 Hz and 30 dB to the raw 30 Hz videos. Methods where we applied Gaussian convolution with $\sigma = 1.5$ are indicated by *. NoRMCorre required Gaussian filtering with at least $\sigma = 0.5$ or would throw an error. Flow-Registration outperforms all other methods in terms of EPE with an improvement between one to two orders of magnitude

Method	Synthetic clean			Synthetic noisy 35 dB			Synthetic noisy 30 dB		
	ch1 + ch2	ch1	ch2	ch1 + ch2	ch1	ch2	ch1 + ch2	ch1	ch2
Flow-Reg.	<u>0.05</u>	0.10	0.07	<u>0.52</u>	0.56	1.04	<u>0.90</u>	0.99	1.64
Flow-Reg. fast	<u>0.14</u>	0.22	0.16	<u>0.59</u>	0.66	1.17	<u>1.06</u>	1.10	1.73
NoRMCorre	—	<u>1.93</u>	2.62	—	<u>1.98</u>	3.09	—	<u>2.34</u>	4.27
NoRMCorre*	—	<u>2.26</u>	3.06	—	<u>2.28</u>	3.23	—	<u>2.43</u>	3.89
elastix GC*	0.23	0.59	<u>0.17</u>	1.71	<u>1.18</u>	2.59	2.93	<u>2.00</u>	3.86
elastix MSE	12.32	45.27	<u>6.55</u>	13.64	11.75	<u>7.04</u>	16.64	<u>14.13</u>	35.79
elastix MSE*	<u>1.41</u>	42.84	3.27	<u>2.87</u>	18.86	3.42	4.89	27.48	<u>4.18</u>
elastix CC	7.25	8.65	<u>6.46</u>	6.53	8.17	<u>6.29</u>	6.41	8.21	<u>6.16</u>
elastix CC*	<u>1.20</u>	6.23	2.57	<u>2.65</u>	6.12	3.57	<u>3.70</u>	6.68	4.09
elastix CC + GC*	0.28	0.53	<u>0.21</u>	1.43	<u>1.25</u>	2.67	2.34	<u>1.99</u>	3.76
elastix MI	6.28	<u>5.59</u>	6.91	5.98	<u>5.62</u>	7.16	<u>5.92</u>	6.29	7.28
elastix MI*	<u>1.42</u>	4.06	3.02	<u>2.75</u>	4.66	4.83	<u>4.07</u>	4.90	5.55
elastix MI + GC*	0.36	0.67	<u>0.34</u>	1.42	<u>1.16</u>	2.66	2.03	<u>1.95</u>	3.66
ANTs ElasticSyN	—	<u>1.38</u>	1.54	—	<u>1.68</u>	2.72	—	<u>1.93</u>	3.15
ANTs ElasticSyN*	—	<u>1.44</u>	1.57	—	<u>1.74</u>	2.63	—	<u>1.96</u>	3.30
ANTs SyN	—	1.51	<u>1.42</u>	—	<u>1.60</u>	2.78	—	<u>2.21</u>	3.27
ANTs SyN*	—	<u>1.50</u>	1.51	—	<u>1.61</u>	2.60	—	<u>2.06</u>	3.34
ANTs SyNCC	—	<u>0.13</u>	0.14	—	<u>2.12</u>	3.22	—	<u>3.26</u>	4.41
ANTs SyNCC*	—	0.19	<u>0.17</u>	—	<u>2.23</u>	3.41	—	<u>3.39</u>	4.52

have potentially more information on the motion whenever disjunct structures are visible in the different image channels, such as in the data in Figure S2A-D. Additionally, the energy functional can naturally incorporate a manual weighting term per channel, such as ROIs, to enforce a low value of the dataterm around important image structures. Because of potentially different SNR properties of the two channels and the likelihood of disjoint structures, we evaluate the subquadratic penalizer on each channel separately which is known as separate robustification.^[20] With a weight function p , a moving image $(f_i)_{i=1}^c$ and a fixed image $(f_i^{\text{ref}})_{i=1}^c$ with c channels, the final dataterm with separate robustification is given by

$$D_{a_{\text{data}}}(w) = \sum_{i=0}^{c-1} p(x, i) \Psi_{a_{\text{data}}} \left(\left(f_{i_{x_1}}(x+w) - f_{i_{x_1}}^{\text{ref}}(x) \right)^2 + \left(f_{i_{x_2}}(x+w) - f_{i_{x_2}}^{\text{ref}}(x) \right)^2 \right) \quad (3)$$

and the smoothness term as

$$S_{a_{\text{smooth}}}(w) = \Psi_{a_{\text{smooth}}} \left(|\nabla u|^2 + |\nabla v|^2 \right) \quad (4)$$

For the numerical approximation, we follow the framework of Bruhn et al.,^[22] Brox et al.^[15] and Papenberg et al.^[23] and discretize the *Euler Lagrange equations* in the compact *motion tensor* notation to solve them with an iterative multiscale solver (downsampling factor $\eta \in [0,1]$) with lagged non-linearities with an update in every 5 iterations. We apply the recommended good practices proposed by Sun et al.,^[14] namely, we use bicubic interpolation for the warping steps and 5×5 median-filtering (mirror boundary) of the flow increments for each level to increase accuracy (see Appendix for algorithm details).

2.5.3 | Pre-processing and registration

Low-pass filtering is an important pre-processing step for local and global methods.^[24] It makes the images derivable and integrates over temporal changes, while removing

image noise. We found convolution with a 3D Gaussian kernel together with subquadratic penalization, to be robust enough to deal with the noise in the benchmark data, for example, compare the layer 5 data (see Figure S2).

In the context of optical microscopy recordings, the recorded image region is usually fixed in the paradigm which means that only little relative movement of the object and camera is expected and the scene content should not change. Therefore, we use a fixed reference frame f_{ref} and estimate the flow field $w_t = (u_t, v_t)^\top$ at time t such that $f(x + u_t, y + v_t, t)$ depicts the same scene points as $f_{\text{ref}}(x, y)$. For the final motion compensation step, we apply backward warping with bilinear interpolation to calculate the motion compensated frame \hat{f} at time t such that $\hat{f}(x, y, t) = f(x + u_t, y + v_t, t)$.

2.5.4 | Parameter selection

The main parameters of our method are the regularization parameter α , the penalizer powers a_{data} and a_{smooth} , the warping depth, the downsampling factor η and for each channel the Gaussian kernel size $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ for pre-processing. For the regularizer, larger values of a_{smooth} reduce discontinuities in the flow field (staircasing artifacts) and for the dataterm, smaller values of a_{data} reduce the influence of outliers and render the method more robust under noise. For 2-Photon recordings, we set $a_{\text{smooth}} = 1$, which results in a homogeneous diffusion regularizer. Note that for $a = 0.5$ we get a regularized ℓ^1 norm, while the function becomes non-convex for values of $a < 0.5$. However, Sun et al. encourage a choice of $a_{\text{data}} = 0.45$ on the Middlebury benchmark, which we apply here. The choice of α presents the compromise of correctly registering smaller structures that deviate from the global motion direction and a globally smooth solution. Using a different value of α in each of the image dimensions renders the smoothness term anisotropic. This might improve the estimation of motion artifacts induced by horizontal scanning.

2.5.5 | Toolbox design and implementation

The ImageJ/FIJI plugin makes use of the ImageJ file formats and therefore can resort to all supported file types. Our software design for the MATLAB toolbox consists of modular file readers and writer classes that can be automatically instantiated or supplied as parameters to an options object that defines registration jobs. By default, MDF, tiff image stacks, MATLAB mat files or hierarchical dataformat (HDF5) files are supported. Additionally,

the output formats support HDF5 files that are readable by the 2-Photon imaging suites Begonia^[25] and CaImAn.^[26] The file IO is designed for multi-channel processing and supplies 4D matrices in the format height \times width \times channel \times time independently of the actual data representation on disk. To compensate a recording, the file reader supplies n subsequent frames (batches of size n) with on-the-fly binning to the Flow-Registration engine until the end of the video is reached. The frames in each batch are concurrently compensated and the average displacement of the last frames is used to initialize the lowest pyramid level of the displacement estimation in the following batch.

Reference frames can either be supplied directly or are computed from a specified set of frames, where the set of frames is pre-aligned with respect to the temporal average and then temporally averaged again. For this pre-alignment, the parameters α and σ are increased in size to make the result more robust under noise and reduce overfitting. Given that each batch can have varying average values, we normalize the data with respect to the minimum and maximum values of the reference frame after applying the Gaussian filter. Joint normalization is performed by default but channel-wise normalization is supported as well

The displacements of each frame with respect to the reference on the lowpass filtered sequences are computed and the raw frames are then registered via backwards warping with bicubic interpolation, where out-of-bounds values are replaced with the values from the reference. To reduce quantization due to the interpolation, the results are stored with double precision as default but can also be stored with the precision of the input file.

To increase computational speed, the finest pyramid level for the OF calculation can be reduced in the options and the resulting low-resolution displacement field will be upsampled to the input image resolution in the final computation step. The Flow-Registration plugin implements this in the *Registration quality* setting, where only the highest quality setting calculates the solution on all levels. With settings $\eta = 0.8$ and minimum level of 6, we get almost tenfold faster computation time on the injection sequence with almost the same accuracy (PSNR 42.523 vs 42.519). With those parameters, the highest resolution at which the displacements are estimated are given by $512 \cdot 0.8^6 = 134$. Batch processing of multiple input files is possible via a batch processor class where a list of filenames is supplied that will be compensated either with individual references or the same reference.

To run the ImageJ/FIJI plugin, it needs to be installed via Plugins \rightarrow Install Plugin which adds a Flow-Registration entry under Plugins. In its current version, the ImageJ/

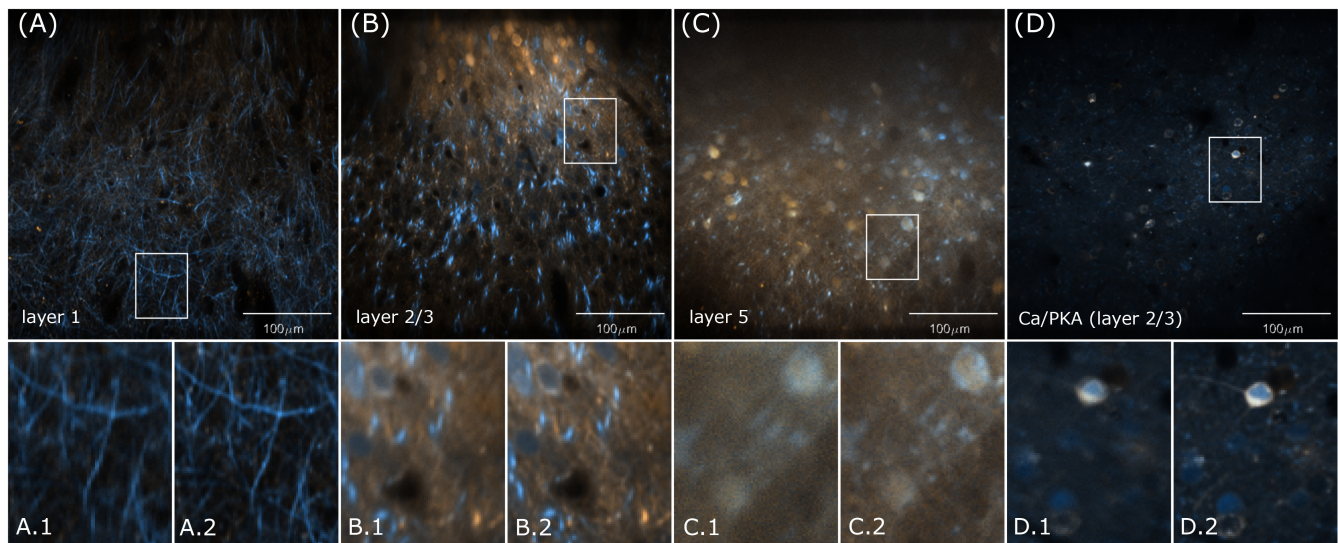


FIGURE 2 The benchmark datasets from layer 1 (A), layer 2/3 (B), layer 5 (C) and the Ca/PKA (layer 2/3) (D) dataset. Temporal average of raw recordings (A.1-D.1) and after application of Flow-Registration (A.2-D.2), both during the motion contaminated frames. Neurons express GAKdYmut (orange) and TurboRFP (blue) (A), GAKdYmut (blue) and TurboRFP (orange) (B-C) and GAKdYmut (orange) and jRGECO (blue) (D)

Fiji plugin can only be used for the compensation of shorter sequences depending on host system memory, the extension for virtual stacks is planned for the future. The MATLAB toolbox does not have such restrictions and can be used for the batch compensation of big data. For the MATLAB toolbox, a C++ compiler is required and the code has been tested for MATLAB R2018a onwards. The folder *demos* in the MATLAB toolbox contains scripts to reproduce the video results and quantitative results presented here as well as examples on how to use the code for different application scenarios. The *jupiter* demo compensates an amateur jupiter recording and demonstrates different aspects of multi-channel tiff and ROI processing as well as a potential application beyond the scope of neuroimaging.

3 | RESULTS

In this section, we first describe the benchmark dataset as well as the metrics we use for quantitative registration performance assessment. We then present the performance of our method in terms of registration quality and computational speed.

3.1 | Benchmark datasets and metrics for evaluation

We used multiple datasets that have been recorded with the described setup previously. The dataset names indicate the imaging depth (see Figures 2 and 4), where

deeper layers (e.g. 5) correspond to lower SNR. For the layer 1–5 recordings, timepoints during movement onset which were contaminated with movement artifacts were selected from continuous recordings. For the datasets during drug and saline injections, we selected the time points around the injection events (e.g. compare Figure 1E). From each dataset, we selected 2500 frames (80.9 s). A common approach for the evaluation of 30 Hz 2-photon microscopy data is temporal binning to increase the SNR. Therefore, our final benchmark dataset contains the data with binning over five frames (6.2 Hz, 500 frames total), as well as on a subset of the raw recordings, indicated by the suffix 30 Hz. For the dataset *saline injection*, in total 5 frames from the beginning and end of the experiment were excluded due to artifacts.

Those data are real-world, low SNR datasets without ground truth displacements, therefore, metrics such as AEE or AAE are not applicable. The same holds true for perception-based metrics due to the overall small movements and high image noise in the data. For the evaluation, we calculate reference-based PSNR (see Table 2) and performance factors based on MSE as well as temporal STD (see Figure 4). To calculate the metrics, we first applied 2D Gaussian convolution with $(\sigma = [3,3]^T)$. As reference, we used the temporal average over the reference frames from the motion compensation. The MSE and STD performance factors are then calculated as the fold change of the average MSE and STD values $\text{MSE}(\text{raw}) \cdot \text{MSE}(\text{compensated})^{-1}$ and $\text{STD}(\text{raw}) \cdot \text{STD}(\text{compensated})^{-1}$. They indicate the factor by which the MSE or STD value is higher on the raw

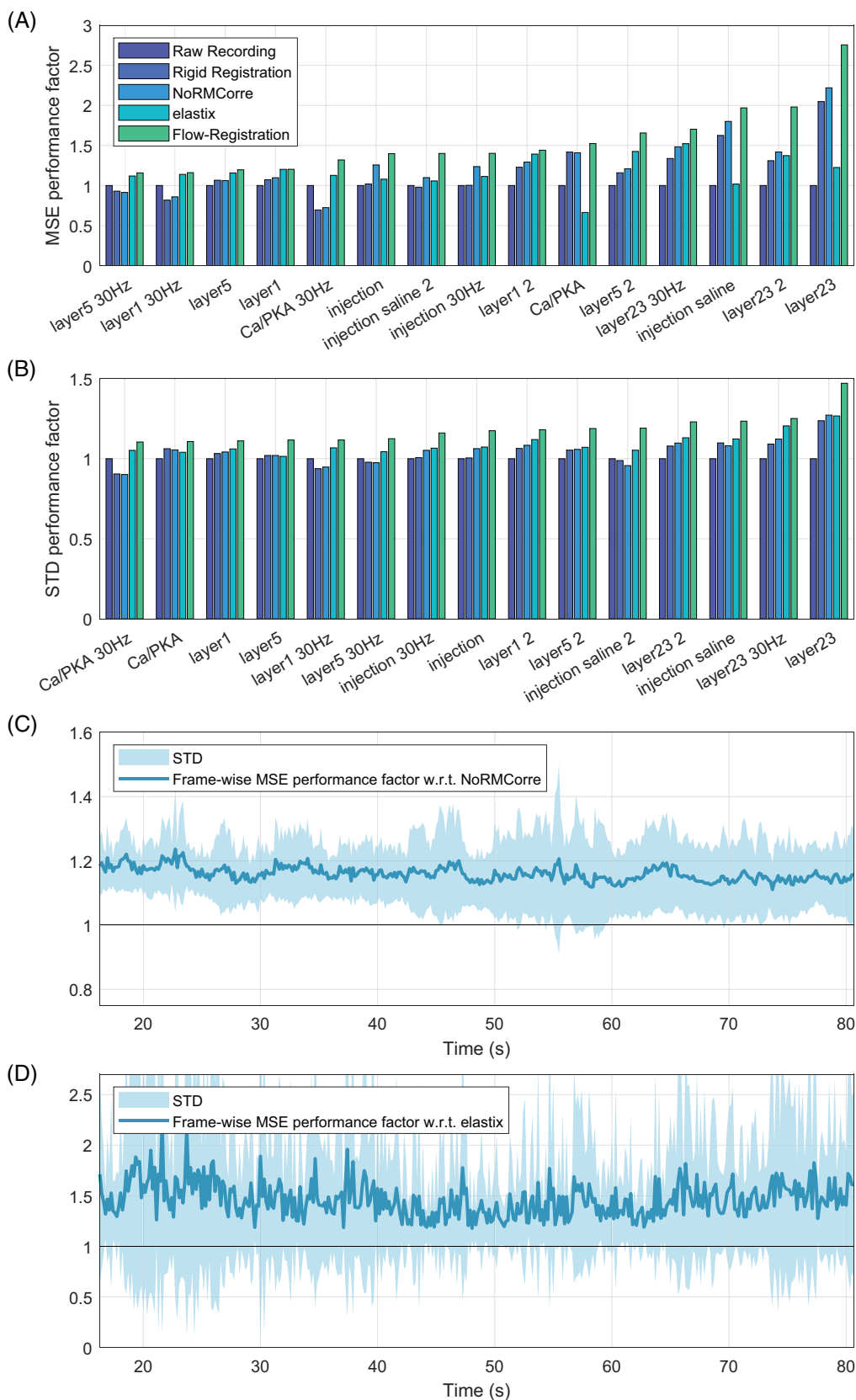


FIGURE 4 Application of Flow-Registration to 15 different datasets. We use 2-Photon recordings from layer 5, layer 2/3, layer 1 and three sequences during in vivo drug or saline injection at 6.2 Hz and 30.9 Hz. In all applied metrics, Flow-Registration performs consistently better than rigid registration and NoRMCorre. The metrics are averaged MSE (A) and STD (B) performance factors (see section 3.1) with respect to the raw recording. (C) displays the frame-wise MSE performance factor of Flow-registration with respect to NoRMCorre over all 6.2 Hz datasets with 500 frames (layer1 - layer 5, injection, injection saline 2; values >1 indicate better performance of Flow-Registration). The frame-wise performance is significantly better for Flow-Registration when compared with NoRMCorre (C) and elastix (D) ($p < 0.00001$, paired, two-sided Wilcoxon signed rank test)

TABLE 2 Average PSNR values (the higher the better) on each dataset for the different methods. To reduce the influence of shot noise and isolate changes due to residual movement, the recordings have been filtered with a 2D Gaussian with $\sigma = 3$ before estimating the PSNR. PSNR has been calculated with respect to the maximum value 2^{16} . The highest PSNR value in each row is put in bold. Flow-Registration consistently outperforms the other methods

Dataset	Raw	Rigid	NoRMCorre	elastix GC*	Flow-Registration
Ca/PKA	86.16	87.57	87.56	84.90	87.75
Ca/PKA 30 Hz	82.09	80.64	80.79	82.60	83.01
injection	38.98	39.60	41.46	39.14	42.52
injection 30 Hz	38.40	38.84	40.25	38.66	41.09
injection saline	78.21	80.30	80.85	77.99	81.34
injection saline 2	78.63	78.61	79.18	78.77	80.16
layer1	91.66	91.87	91.96	92.36	92.37
layer1 2	87.53	88.13	88.31	88.56	88.69
layer1 30 Hz	85.13	84.26	84.44	85.69	85.76
layer23	77.14	79.47	79.81	77.82	80.79
layer23 2	89.24	90.24	90.60	90.30	91.87
layer23 30 Hz	73.25	74.30	74.79	74.89	75.45
layer5	90.11	90.34	90.32	90.70	90.83
layer5 2	88.73	89.16	89.29	90.01	90.6
layer5 30 Hz	83.76	83.45	83.38	84.24	84.39

sequence than the MSE or STD of each method (see Figure 4 (A) and (B)) or the factor by which the MSE value is higher for NoRMCorre when compared to Flow-Registration on a frame-wise basis (see Figure 4 [C]). We excluded all frames that contributed to the reference, as well as a boundary of 25 pixels. This boundary ensures that only valid regions are considered in the evaluation even with the large displacements of the injection sequences. Lowpass filtering reduced the influence of image noise on the results. For the compensation of all sequences, the first 100 frames (6.2 Hz) and 500 frames (30.9 Hz) were used as reference and supplied to the respective method.

We construct a pair of synthetic frames from the real world recording that match their statistics with synthetic ground truth displacements and noise. On this synthetic data, we compare the average *endpoint error* (EPE) for Flow-Registration, NoRMCorre as well as elastix and ANTs with different configurations.

For this purpose, we use the injection sequence and add simulated injection, xyz-shifts as well as noise to the clean image we obtain after temporal averaging of the motion compensated recording. We compose the synthetic motion field of a component for line jitter from the scanning as well as high divergence displacements to simulate injection. The injection displacements are modeled as (in pixel coordinates with 512×512 resolution)

$$w_{inj}(x,y) = -(256, 280)^\top + 0.05 \cdot \begin{cases} (x,y)^\top, & \text{if } y \geq 280 \\ (x, 0.2 \cdot y)^\top, & \text{otherwise} \end{cases}$$

For the jitter, we apply $w_{jit}(x,y) = (\sin(0.001x\pi), 0)^\top$ which results in the final displacement $w = w_{inj} + 2w_{jit}$ (see Figure 3B, left). We adapt the noise model and implementation of Poisson image noise for 2-photon microscopy from Zhang et al.^[27] and apply noise with target PSNR of 30 dB and 35 dB. To simulate changes in the intensity of the visible structures for example due to the injected indicators or due to z-shifts, we apply positive and negative multiplicative intensity changes with a scaled, 2D Gaussian kernel. See Figure 3A for a comparison of the benchmark frames and the original video.

3.2 | Experiments with synthetic data

We compare our method with the 2-Photon motion compensation suite NoRMCorre and with the traditional image registration toolboxes ANTs and elastix. For ANTs and elastix, we use the respective Python (version 3.9.7) implementations, namely, *antspy* (version 0.3.1) and *itk-elastic* (itk version 5.2.1, itk-elastic version 0.13.0) and for NoRMCorre we use the official MATLAB implementation. Evaluations are performed with a python interpreter on the *Windows subsystem for linux* (WSL) 2 (hardware acceleration). ANTs and elastix allow for similar choices

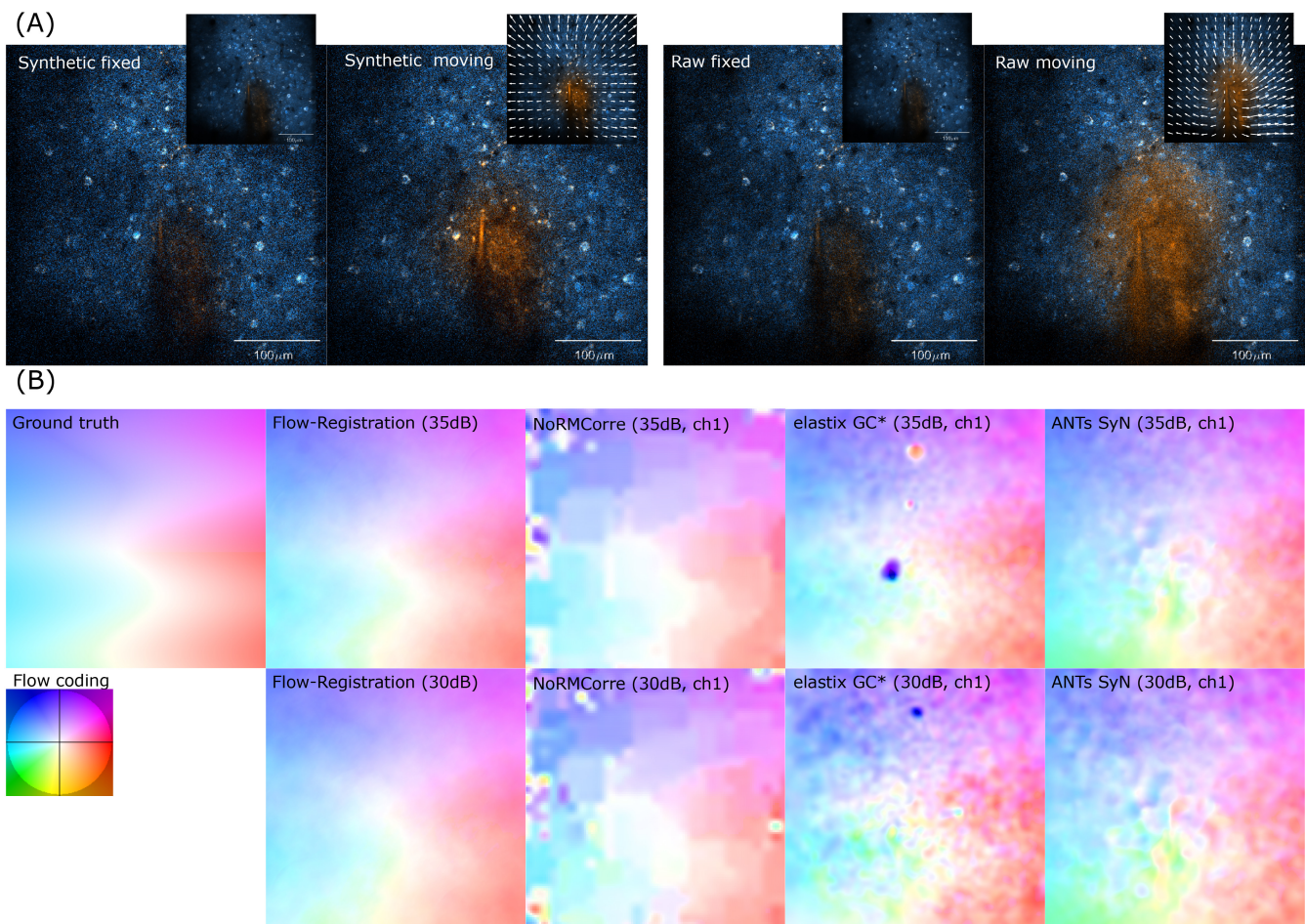


FIGURE 3 The two frames from the synthetic benchmark (A, left) with 35 dB PSNR in comparison with raw frames from the 6 Hz injection data (A, right). The small images show the clean frames without motion and with the synthetic displacement field (A, left) and the estimated displacement field (A, right). The clean frames of the real recording have been calculated by averaging frames from the motion compensated injection sequence. (B) shows the color-coded displacement fields as estimated by each method alongside the ground truth displacement. We report the channel configuration that performed best (see Table 1). Reported run times might be better for native ANTs and elastix implementations. NoRMCorre and Flow-Registration have been evaluated with the same system and software configuration

of dataterms and transformations. While elastix supports extensive registration configurations in config files that are backed by a model zoo, ANTs has multiple presets and allows easy adaptation of the registration method. For elastix, we report the results with different metrics and simulated gradient constancy. For ANTs, we report the performance of the default *SyN* presets with *cross-correlation* (CC) and *mutual information* (MI) metric. We report the results on the individual channels for all methods and on multichannel configurations for our method as well as for elastix. For elastix, we simulated gradient constancy as a multichannel input on the up to four gradient channels (directional gradients in x and y direction on both channels) with *AdvancedMeanSquares* metric. The results indicate, that gradient constancy is superior over other dataterms. With lower PSNR, subquadratic penalization becomes vital,

such that Flow-registration outperforms all other methods — for some methods more than an order magnitude. Our results also indicate that for users of elastix, it could make sense to use MI and CC on the gradients of the images in certain situations.

The evaluations confirm our assumption, that the combination of two channels can improve the performance on noisy data: Our method performs consistently better with both channels as compared to evaluations on the individual channels. The same is not true for elastix and ANTs, indicating that there are parameters such as metric weights that could be further optimized for an overall better performance. While some of the traditional image registration methods implemented in elastix outperform NoRMCorre in terms of EPE, NoRMCorre's performance seems to be relatively robust under noise and does not benefit from Gaussian pre-smoothing which

confirms the results from their paper.^[5] However, their performance cannot match Flow-Registration, which is up to a factor 38 better on the clean recording and still by a factor of 2.6 on the noisy 30 dB recording.

In terms of running time, we cannot directly compare our method with elastix and ANTs due to the fact that the evaluations have been done on an emulated environment. Qualitatively, all evaluations of the other methods were significantly slower than Flow-Reg. fast which took around 0.5 s for the complete estimation without parallelization.

3.3 | Experiments on real recordings

For the real world evaluation, we include our method, NoRMCorre as well as elastix with gradient constancy and MSE metric. For NoRMCorre, we calculate the displacements for each channel individually and for elastix, we use the joint estimation because the EPE on both channels was lower than the average EPE of the separate channels.

Flow-Registration consistently outperforms elastix,^[8] NoRMCorre^[5] and rigid image registration on the benchmark datasets in terms of the reduced temporal STD and MSE ratios (see Section 3.1) as well as *peak signal to noise ratio* (PSNR), see Figure 4 and Table 2. Frame-wise comparisons of frames registered with Flow-Registration and NoRMCorre as well as with elastix show a significantly higher ($p < 0.00001$, paired, two-sided Wilcoxon signed rank test) performance of Flow-Registration, see Figure 4C,D. The graph indicates the factor, by which the NoRMCorre MSE is higher than the Flow-Registration MSE (i.e. >1 indicates better performance of our method). While NoRMCorre has a better frame-wise MSE performance on the 6 Hz recordings than elastix on average, the difference was not found to be significant ($p > 0.05$, paired, two-sided Wilcoxon signed rank test). Qualitatively, our method produces much crisper average frames compared to other methods during challenging motion events such as local drug injections with large intensity changes in the different channels, see Figure 1 and supplementary figures.

For a fast approximation of the solution, the minimum warping depth can be used to define the maximum resolution at which the solution is computed. We found around 10x speedup to be possible without reducing compensation quality (avg PSNR 42.523 with the precise and 42.519 with the approximated solution, min level = 6).

In terms of computation time, the MATLAB toolbox is similar to existing methods and even faster with the approximated solution. On a single channel version of the 500 frame injection sequence (consumer workstation, 12 cores @ 3.8 GHz, 64GB of memory), our method (50

iterations, no pre-processing/IO, min level 0) takes 115 s and only 15 s with min level 6, while NoRMCorre (grid size 32, one iteration) takes 160 s and with grid size 16 even 763 s (setting used for the benchmark solution).

Flow-Registration was run for all datasets with default parameters and $\alpha = 1.5$, for the 6.2 Hz datasets with $\sigma = (1,1,0.1)^T$ and on the 30.9 Hz datasets with $\sigma = (1.5,1.5,0.5)^T$. For the injection sequence, additionally the channel weight was set to $(1.15,0.85)^T$. The batch size was set to the size of the datasets.

4 | DISCUSSION

We have presented the new motion compensation method Flow-Registration and can report state-of-the-art registration results. The MATLAB implementation outperforms NoRMCorre ANTs and elastix in terms of registration quality as well as computation speed with comparable parameters. NoRMCorre is an established 2-photon neuroimaging motion compensation method and the default method in recent 2-Photon and calcium imaging suites such as CaImAn (2019),^[26] EZcalcium (2020)^[28] or Begonia (2021)^[25] and can be considered the state of the art for the alignment of 2-photon videos. Our toolbox can produce output files that are compatible with CaImAn as well as with Begonia and can be easily extended for other frameworks and workflows via object oriented IO file readers and writers. ANTs and elastix are intensity based image registration methods which implement different dataterms and metrics for mono- and multimodal image registration.^[8–11] Our method outperforms ANTs and elastix by at least factor 2 in terms of EPE on the synthetic data. Flow-Registration also shows a consistently higher performance on the real world recordings than elastix. The best performing elastix configurations were simulated gradient constancy with Gaussian smoothing which is close to the Optimization target in Flow-Registration but without a subquadratic penalizer. Elastix seems to be less robust under changes in high frequent jitter or motion blur resulting in conflicting performance on the real world datasets before and after binning. On some of the real world recordings, elastix shows a better performance than NoRMCorre; however, the frame-wise performance difference for NoRMCorre and elastix was not found to be significant on the 6 Hz recordings. Pnevmatikakis et al. found that registration on structural channels does not necessarily result in better registration performance when compared to registration on the functional channel.^[5] While we also could not identify a clear trend on the synthetic data, our evaluations show that the joint optimization of a structural and functional channel consistently outperformed the

single channel configurations for Flow-registration. A potential explanation could be an effective reduction in the SNR for joint structures as well as a higher amount of structural information in the presence of disjoint structures in the different channels. Those effects could be even more relevant for multispectral images with a higher number of channels.

The evaluations on synthetic data have shown that the current state of the art can be outperformed with metrics and regularizers common in variational OF estimation. Particularly, the experiments presented here could demonstrate the benefit of a sub-quadratic penalizer of the image gradients for low SNR 2-Photon imaging data which can perform better than metrics with more invariances under gray value transformations such as mutual information or cross-correlation based metrics. This also suggests that the implementation of subquadratic penalizers and gradient or higher order based constancies could be a useful extension to other image registration toolboxes such as elastix and ANTs.

Flow-Registration reliably corrects movement artifacts even when mixed with intensity changes, as local calcium signals, spreading calcium waves, or calcium responses triggered by local drug injections. The method will record all movements also those of physiological origin, for example, the constriction or dilation of blood vessels. While this implies that those changes cannot be analyzed in the compensated recording, the complete motion information is contained in the estimated displacement fields. Decomposition of the displacements into movements with different transforms or fixed with respect to certain points could enable the selective visualization of movement components of interest for example with Lagrangian motion magnification^[29] as well as quantification of those changes.

Motion compensation methods often incorporate many regularization parameters that require difficult fine tuning for different motion statistics. For example, NoRMCorre has a total of 7 regularization and 2 sub-pixel refinement parameters.^[5] A limitation of Flow-Registration as well as of the compared methods is the influence of the parameters on the results making the parameter choice a crucial step in the pipeline. However, our method needs only a single regularization parameter and one parameter for pre-smoothing, while the results have native subpixel precision because of the continuous modeling. We have extended the concept for line scan alignment from our previous work^[6] to the compensation of recordings with narrow FOV by implementing non-uniform warping that allows more pyramid steps along the larger dimension of the recording.

Generally, a drawback of 2D motion compensation approaches for 2-photon imaging is the lack of z-shift correction. While there exist methods for high-speed, online 3D compensation,^[30] they require complicated setups and the current generation is limited to rigid 3D motion compensation — which might therefore benefit from refinement with a method such as Flow-Registration as well.

While our method does not aim for online processing, deep learning-based approaches could unlock real time applications of our method. Even state-of-the-art self-supervised methods often require annotated data as initialization.^[31] Flow-Registration has already been applied to many state-of-the-art 2-photon imaging recordings. The explicit, high-accuracy estimation of displacements with our method can be used for the generation of datasets for the training of efficient, deep learning-based motion compensation methods in the future.

5 | CONCLUSION

The solutions presented in this paper solve the pre-processing problem of motion contamination in 2-photon microscopy and multichannel video recordings with difficult, non-linear motion. Our evaluations show that sub-quadratic metrics with gradient constancy and homogeneous smoothness can outperform methods with more advanced data invariances such as given by CC or MI constancies and deformations on this data, while, at the same time, it is easier to optimize as demonstrated in the OF literature. We implemented this method into an accessible MATLAB toolbox and ImageJ plugin.

The core software design paradigm of our toolbox was the easy yet versatile integration into different workflows and toolboxes for 2-Photon imaging. We have developed a MATLAB toolbox that supports common file formats such as MDF, tiff image stacks, MATLAB mat files or HDF5 files in single file or multichannel configurations. The image IO is designed in a modular, object oriented way, such that the toolbox can easily be extended with new data formats and embedded pre-processing. The code is memory efficient and allows to compensate bigdata recordings with specified pre-processing methods and on-the-fly binning. The computationally heavy code is written in C++ which allows the implementation of Python wrappers in the future. The ImageJ/FIJI plugin builds on Imglib2^[32] library which supports most common image formats through the bio-formats plugin. The implementations support weighted, multichannel input with weight masks to enforce higher weight on the dataterm inside of ROIs. The ImageJ/FIJI plugin is integrated with the MATLAB toolbox, such that the plugin can export parameters and

reference frame configurations, that can be loaded as MATLAB bulk registration jobs.

Documentation, the MATLAB code for Flow-Registration and the ImageJ Plugin can be found on the GitHub repository https://github.com/phfloth/flow_registration. The version used for the evaluations in this paper is supplied as supplemental code. It contains MATLAB scripts to reproduce the results reported here as well as the precompiled ImageJ/FIJI plugin.

AUTHOR CONTRIBUTIONS

Philipp Flotho contributed to software, methodology, conceptualization, data curation, formal analysis, visualization, and writing-original draft. **Shinobu Nomura** contributed to data curation, investigation, writing-review & editing, and validation. **Bernd Kuhn** contributed to conceptualization, writing-review & editing, supervision, and resources. **Daniel J. Strauss** contributed to conceptualization, writing-review & editing, and supervision.

ACKNOWLEDGMENTS

This work was partially funded by the Japan Society for the Promotion of Science (JSPS) with the Summer Program 2019 under grand number SP19305 and partially conducted at the *Optical Neuroimaging Unit* under Bernd Kuhn at the Okinawa Institute of Science and Technology Graduate University. The authors thank Miles J. Desforges, Mohamed Eltabbal and Lars Haab for code testing and valuable feedback and discussions. Open Access funding enabled and organized by Projekt DEAL.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

DATA AVAILABILITY STATEMENT

The complete benchmark dataset used in this work is available as 2-Photon Movies with Motion Artifacts on Dryad.

ORCID

Philipp Flotho  <https://orcid.org/0000-0002-8480-0085>

Shinobu Nomura  <https://orcid.org/0000-0001-8883-5405>

Bernd Kuhn  <https://orcid.org/0000-0002-6852-2433>

Daniel J. Strauss  <https://orcid.org/0000-0001-8481-499X>

REFERENCES

- [1] F. Helmchen, W. Denk, *Nat. Methods* **2005**, 2(12), 932.
- [2] P. Theer, B. Kuhn, D. Keusters, W. Denk, *Encyclop. Mol. Cell Biol. Mol. Med* **2006**, 15, 61.
- [3] B. K. P. Horn, B. G. Schunck, *Artif. Intell.* **1981**, 17(1–3), 185.
- [4] D. S. Greenberg, J. N. D. Kerr, *J. Neurosci. Methods* **2009**, 176(1), 1.
- [5] E. A. Pnevmatikakis, A. Giovannucci, *J. Neurosci. Methods* **2017**, 291, 83.
- [6] P. Flotho, D. Thinnies, B. Kuhn, C. J. Roome, J. F. Vibell, D. J. Strauss, *J. Neurosci. Methods* **2021**, 353, 109076.
- [7] C. J. Roome, B. Kuhn, *elife* **2020**, 9, e59619.
- [8] S. Klein, M. Staring, K. Murphy, M. A. Viergever, J. P. W. Pluim, *IEEE Trans. Med. Imaging* **2009**, 29(1), 196.
- [9] B. B. Avants, N. Tustison, G. Song, et al., *Insight j* **2009**, 2(365), 1.
- [10] B. B. Avants, C. L. Epstein, M. Grossman, J. C. Gee, *Med. Image Anal.* **2008**, 12(1), 26.
- [11] B. B. Avants, N. J. Tustison, G. Song, P. A. Cook, A. Klein, J. C. Gee, *NeuroImage* **2011**, 54(3), 2033.
- [12] Z. Chen, H. Jin, Z. Lin, S. Cohen, Y. Wu, *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit*, IEEE Computer Society Press, New York **2013**, p. 2443.
- [13] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, R. Szeliski, *Int. J. Comput. Vis.* **2011**, 92(1), 1.
- [14] D. Sun, S. Roth, M. J. Black, *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit*, IEEE Computer Society Press, New York **2010**, p. 2432.
- [15] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, *Comput Vis ECCV*, Springer, Berlin **2004**, p. 25.
- [16] A. Bonnot, E. Guiot, R. Hepp, L. Cavellini, L. Tricoire, B. Lambomez, *FASEB J.* **2014**, 28(3), 1375.
- [17] S. Nomura, L. Tricoire, I. Cohen, B. Kuhn, B. Lambomez, R. Hepp, *iScience* **2020**, 23(11), 101710.
- [18] C. J. Roome, B. Kuhn, *Front. Cell. Neurosci.* **2014**, 8, 379.
- [19] C. J. Roome, B. Kuhn, *Multiphoton Microscopy*, Springer, New York **2019**, p. 297.
- [20] H. Zimmer, A. Bruhn, J. Weickert, L. Valgaerts, A. Salgado, B. Rosenhahn, H.-P. Seidel, *Energy Minimization Methods Comput Vis Pattern Recognit*, Springer, Berlin **2009**, p. 207.
- [21] G. Hermosillo, C. Chef d'Hotel, O. Faugeras, *Int. J. Comput. Vis.* **2002**, 50(3), 329.
- [22] Andrés Bruhn, Ph.D. thesis, Saarland University, **2006**.
- [23] N. Papenberg, A. Bruhn, T. Brox, S. Didas, J. Weickert, *Int. J. Comput. Vis.* **2006**, 67(2), 141.
- [24] A. Bruhn, J. Weickert, C. Schnörr, *Int. J. Comput. Vis.* **2005**, 61(3), 211.
- [25] D. M. Bjørnstad, K. S. Åbjørsbråten, E. Hennestad, C. Cunen, G. H. Hermansen, L. Bojarskaite, K. H. Pettersen, K. Vervaeke, R. Enger, *Front. Cell. Neurosci.* **2021**, 15, 176.
- [26] A. Giovannucci, J. Friedrich, P. Gunn, J. Kalfon, B. L. Brown, S. A. Koay, J. Taxisidis, F. Najafi, J. L. Gauthier, P. Zhou, B. S. Khakh, D. W. Tank, D. B. Chklovskii, E. A. Pnevmatikakis, *elife* **2019**, 8, e38173.
- [27] Y. Zhang, Y. Zhu, E. Nichols, Q. Wang, S. Zhang, C. Smith, S. Howard, *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit*, IEEE Computer Society Press, New York **2019**, p. 11710.
- [28] D. A. Cantu, B. Wang, M. W. Gongwer, C. X. He, A. Goel, A. Suresh, N. Kourdougli, E. D. Arroyo, W. Zeiger, C. Portera-Cailliau, *Front. Neural. Circuits* **2020**, 14, 25.

- [29] P. Flotho, M. J. Bhamborae, L. Haab, D. J. Strauss, *Annu Int Conf IEEE Eng Med Biol Soc, IEEE, IEEE, Piscataway, NJ* **2018**, p. 3586.
- [30] V. A. Griffiths, A. M. Valera, J. Y. N. Lau, H. Roš, T. J. Younts, B. Marin, C. Baragli, D. Coyle, G. J. Evans, G. Konstantinou, et al., *Nat. Methods* **2020**, *17*(7), 741.
- [31] P. Liu, M. Lyu, I. King, J. Xu, *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit*, IEEE Computer Society Press, New York **2019**, p. 4571.
- [32] T. Pietzsch, S. Preibisch, P. Tomančák, S. Saalfeld, *Bioinformatics* **2012**, *28*(22), 3009.

SUPPORTING INFORMATION

Additional supporting information may be found in the online version of the article at the publisher's website.

How to cite this article: P. Flotho, S. Nomura, B. Kuhn, D. J. Strauss, *J. Biophotonics* **2022**, *15*(8), e202100330. <https://doi.org/10.1002/jbio.202100330>