Robust and Efficient Alignment of Calcium Imaging Data through Simultaneous Low Rank and Sparse Decomposition - Supplementary materials

Junmo Cho[1]*, Seungjae Han[1]*, Eun-Seo Cho[1], Kijung Shin[1,2], Young-Gyu Yoon[1,3]
[1]School of Electrical Engineering, KAIST
[2]Kim Jaechul Graduate School of AI, KAIST
[3]KAIST Institute for Health Science and Technology
Daejeon, Republic of Korea
{junmokane, jay0118, eunseo.cho, kijungs, ygyoon}@kaist.ac.kr

1. Supplementary Notes

1.1. Larval zebrafish brain dataset description

Confocal microscopy images of larval zebrafish brains were acquired using a point-scanning confocal microscopy system (C2 Plus, Nikon) equipped with a 16x 0.8NA water dipping objective lens. We acquired 2-D video of neural activity at specific depths and 3-D video of neural activity by concatenating 2-D videos at different depths. For 2-D video, we obtained images with a size of $512(x) \times 512(y)$, and 60 frames in time. For 3-D video, we obtained volume images with a size of $256(x) \times 512(y) \times 48(z)$, and 60 frames in time.

For zebrafish experiments, larval zebrafish expressing pan-neuronal GCaMP7a with a casper background were imaged at 4 days post-fertilization (dpf). The fish were paralyzed by immersing in $0.25mg/ml$ of pancuronium bromide (Sigma-Aldrich) solution for 2 minutes and then embedded in agar with 2.0% low melting point agarose (TopVision) in Petri dishes. After the solidification of the agarose gel, the dishes were filled with standard fish water.

1.2. REALS algorithm

Algorithm 1 REALS

Input: $Y \in \mathbb{R}^{m \times n}$
Parameter: $W \in \mathbb{R}^{m \times r}$,
initialize τ^θ as τ^I (I: identity).
η_W, η_θ: learning rate for W, θ respectively.
c: clamping constant.
Output: $\tau^\theta(Y), L, S \in \mathbb{R}^{m \times n}$

while not done **do**
  for $i = 1$ to r **do**
    $L_i = \sum_j < W_j, \tau_i^\theta(Y_i) > W_j$
    $S_i = \tau_i^\theta(Y_i) - L_i$
  end for
  Update $W \leftarrow W - \eta_W \frac{\partial ||S||_1}{\partial W}$
  Update $\theta \leftarrow \theta - \eta_\theta \frac{\partial ||S||_1}{\partial \theta}$
  $\theta \leftarrow \text{clamp}(\theta, I - c, I + c)$
end while
return $\tau(Y), L, S$

*Equal contribution

2. Supplementary Movies

- **Movie S1** Examples of aligned images and the corresponding *MSE*. Images of larval zebrafish brain were synthetically perturbed then aligned. *MSE*; 1st column: 0.000015, 0.00025, 0.0004 (from left), 2nd column: 0.001, 0.003, 0.006 (from left)

- **Movie S2** Alignment of images with additional image corruption using REALS. Images of larval zebrafish brain were synthetically corrupted, perturbed then aligned. 1st column: input images, 2nd column, aligned images, 3rd column: recovered low rank component, 4th column: sparse component. (a) With Poisson noise (brightness level of 100). (b) With Gaussian noise ($\sigma = 0.01$). (c) With reduced scale of low rank component ($\alpha = 0.4$).

- **Movie S3** Alignment of images with additional deformation using deformable REALS and Voxelmorph. 1st column: Input images, REALS, Voxelmorph (from left), 2nd column: input images, REALS, Voxelmorph (from left).

- **Movie S4** Alignment of a mouse brain image dataset that suffers from large motion. (a) Input images. (b) Aligned images.

- **Movie S5** Alignment of 3-D images using REALS. Three-dimensional images of a larval zebrafish brain were synthetically perturbed then aligned. Top left, lateral projection. Top right, axial projection. Bottom, rostral-caudal projection. (a) Input images. (b) Aligned images. (c) Recovered low rank component. (d) Sparse component. Scale bar, 100 μm.

- **Movie S6** Alignment of larval zebrafish brain images with drift. (a-c) Low rank and sparse decomposition using REALS, but without updating transformation parameters. (a) Input images. (b) Low rank component of **a**. (c) Sparse component of **a**. (d-f) Low rank and sparse decomposition using REALS. (d) Aligned images of **a**. (e) Low rank component of **d**. (f) Sparse component of **d**. Scale bar, 30 μm.

3. Supplementary Figures

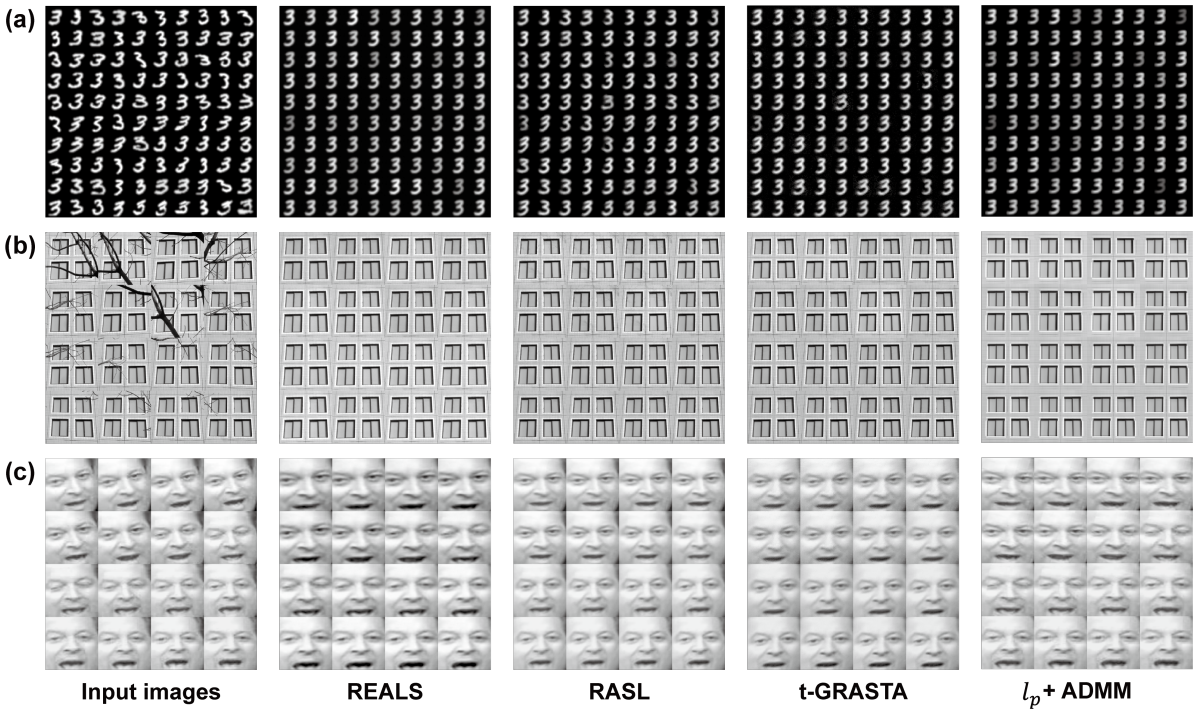

Figure 1: Input images and the recovered low rank images using REALS, RASL, t-GRASTA and l_p+ADMM. (a) 'Digit 3' dataset. (b) 'Window' dataset. (c) 'Al Gore' dataset.

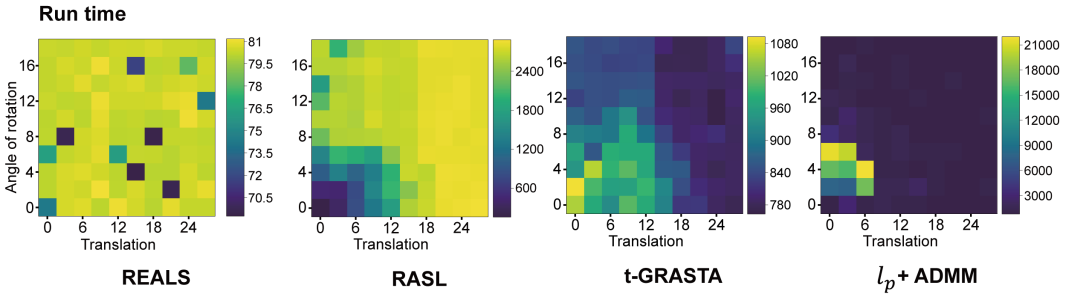

Figure 2: Heatmaps showing the run time of REALS, RASL, t-GRASTA, and l_p+ADMM. The algorithms were tested for the datasets with various levels of geometric perturbations (translation and rotation).

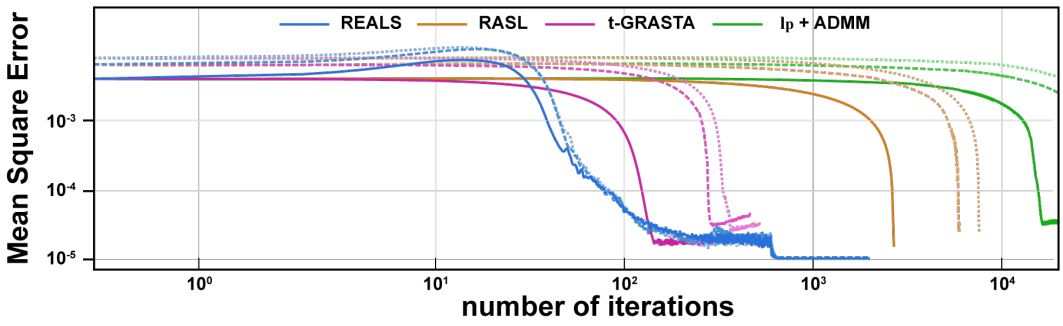

Figure 3: Mean square error of REALS, RASL, t-GRASTA, l_p+ADMM versus number of iterations. Solid: $(t_0, \theta_0) = (12, 8)$, Dashed: $(t_0, \theta_0) = (18, 12)$, Dotted: $(t_0, \theta_0) = (24, 16)$.

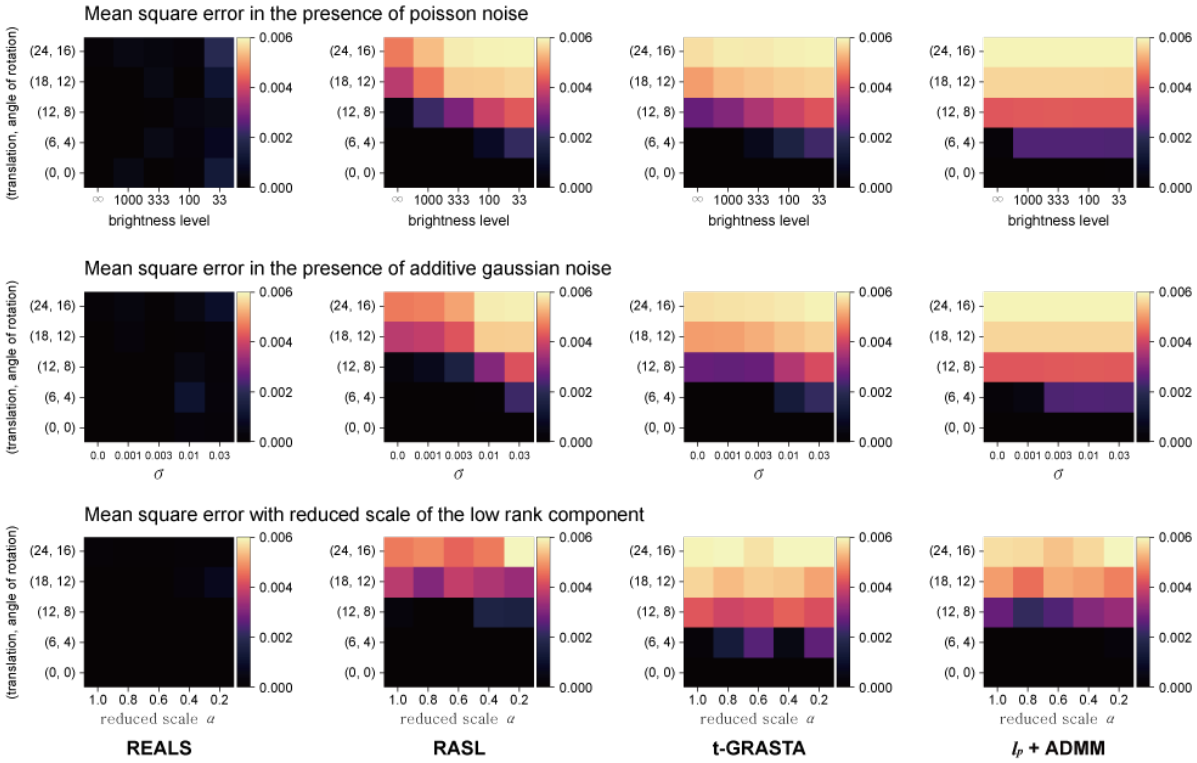

Figure 4: Heatmaps showing the mean squared error of REALS, RASL, t-GRASTA and lp+ADMM. The algorithms were tested for the datasets with various levels of geometric perturbations (translation and rotation) and image corruptions.

4. Supplementary Tables

Table 1: Hyperparameters used for the experiments in section **Translation and Rotation**, and section **Additional image corruption**

hyper-parameter	setting
rank	1
learning rate of W	1×10^{-4}
learning rate of τ	1×10^{-2}
epoch	2000
clamp	$-0.6 \leq \cos(\theta_{11}), \sin(\theta_{12}) \leq 0.6, -0.25 \leq \theta_{13}, \theta_{23} \leq 0.25$
transformation	Euclidean
batch size	60
scheduler	StepLR with step size $600, \gamma = 0.1$

Table 2: Hyperparameters used for the experiments in section **Multi-resolution REALS**. Same parameters were used for REALS and multi-resolution REALS .

hyper-parameter	setting
rank	1
learning rate of W	1×10^{-4}
learning rate of τ	1×10^{-2}
epoch	10000
clamp	$-0.6 \leq \cos(\theta_{11}), \sin(\theta_{12}) \leq 0.6, -0.25 \leq \theta_{13}, \theta_{23} \leq 0.25$
transformation	Euclidean
batch size	60
K	3 (for multi-resolution REALS only)
β_k	4^k (for multi-resolution REALS only)

Table 3: Deformable REALS and Voxelmorph performance

	$\sigma_1 = 0.2, \sigma_2 = 40$		$\sigma_1 = 1, \sigma_2 = 40$	
(t_0, θ_0)	Deformable REALS	Voxelmorph	Deformable REALS	Voxelmorph
$(0, 0°)$	6.40×10^{-5}	1.33×10^{-4}	1.32×10^{-4}	2.48×10^{-4}
$(3, 2°)$	8.21×10^{-5}	2.98×10^{-4}	1.32×10^{-4}	3.94×10^{-4}
$(6, 4°)$	1.39×10^{-4}	7.18×10^{-4}	1.85×10^{-4}	8.68×10^{-4}
$(9, 6°)$	2.42×10^{-4}	8.93×10^{-4}	2.95×10^{-4}	9.45×10^{-4}
$(12, 8°)$	5.23×10^{-4}	1.44×10^{-3}	5.32×10^{-4}	1.53×10^{-3}

Table 4: Hyperparameters used for the experiments in section **Mouse brain dataset**.

hyper-parameter	setting
rank	2
learning rate of W	1×10^{-4}
learning rate of τ	1×10^{-4}
epoch	4000
clamp	$-0.6 \leq \cos(\theta_{11}), \sin(\theta_{12}) \leq 0.6, -0.25 \leq \theta_{13}, \theta_{23} \leq 0.25$
transformation	Euclidean
batch size	1000
scheduler	not used

Table 5: Hyperparameters used for the experiments in section **REALS on 3-D data**.

hyper-parameter	setting
rank	1
learning rate of W	1×10^{-4}
learning rate of τ	1×10^{-3}
epoch	1000
clamp	$\theta_{11}, \theta_{22}, \theta_{33} \leq 1.02$
transformation	affine
batch size	30
scheduler	not used

Table 6: Training details for section **REALS with NMF**.

hyper-parameter	setting
rank	1
number of components	50
learning rate of W	1×10^{-4}
learning rate of τ	1×10^{-3}
epoch	600
clamp	$-0.05 \leq \cos(\theta_{11}), \sin(\theta_{12}) \leq 0.05, -0.05 \leq \theta_{13}, \theta_{23} \leq 0.05$
transformation	Euclidean
batch size	128
scheduler	not used