



HAL
open science

Tangent space alignment: Transfer learning for Brain-Computer Interface

Alexandre Bleuzé, Jérémie Mattout, Marco Congedo

► **To cite this version:**

Alexandre Bleuzé, Jérémie Mattout, Marco Congedo. Tangent space alignment: Transfer learning for Brain-Computer Interface. *Frontiers in Human Neuroscience*, 2022, 16, pp.1049985. 10.3389/fnhum.2022.1049985 . hal-04149663v2

HAL Id: hal-04149663

<https://hal.science/hal-04149663v2>

Submitted on 3 Jul 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



OPEN ACCESS

EDITED BY

Chang-Hwan Im,
Hanyang University, South Korea

REVIEWED BY

Fangzhou Xu,
Qilu University of Technology, China
Jane Zhen Liang,
Shenzhen University, China

*CORRESPONDENCE

Marco Congedo
marco.congedo@gipsa-lab.fr

SPECIALTY SECTION

This article was submitted to
Brain-Computer Interfaces,
a section of the journal
Frontiers in Human Neuroscience

RECEIVED 21 September 2022

ACCEPTED 11 November 2022

PUBLISHED 02 December 2022

CITATION

Bleuzé A, Mattout J and Congedo M
(2022) Tangent space alignment:
Transfer learning for Brain-Computer
Interface.
Front. Hum. Neurosci. 16:1049985.
doi: 10.3389/fnhum.2022.1049985

COPYRIGHT

© 2022 Bleuzé, Mattout and Congedo.
This is an open-access article
distributed under the terms of the
[Creative Commons Attribution License
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or
reproduction in other forums is
permitted, provided the original
author(s) and the copyright owner(s)
are credited and that the original
publication in this journal is cited, in
accordance with accepted academic
practice. No use, distribution or
reproduction is permitted which does
not comply with these terms.

Tangent space alignment: Transfer learning for Brain-Computer Interface

Alexandre Bleuzé¹, Jérémie Mattout² and Marco Congedo^{1*}

¹GIPSA-Lab, University Grenoble Alpes, CNRS, Grenoble INP, Grenoble, France, ²Lyon Neuroscience Research Center, INSERM, CNRS, University Claude Bernard Lyon 1, Lyon, France

Statistical variability of electroencephalography (EEG) between subjects and between sessions is a common problem faced in the field of Brain-Computer Interface (BCI). Such variability prevents the usage of pre-trained machine learning models and requires the use of a calibration for every new session. This paper presents a new transfer learning (TL) method that deals with this variability. This method aims to reduce calibration time and even improve accuracy of BCI systems by aligning EEG data from one subject to the other in the tangent space of the positive definite matrices Riemannian manifold. We tested the method on 18 BCI databases comprising a total of 349 subjects pertaining to three BCI paradigms, namely, event related potentials (ERP), motor imagery (MI), and steady state visually evoked potentials (SSVEP). We employ a support vector classifier for feature classification. The results demonstrate a significant improvement of classification accuracy, as compared to a classical training-test pipeline, in the case of the ERP paradigm, whereas for both the MI and SSVEP paradigm no deterioration of performance is observed. A global 2.7% accuracy improvement is obtained compared to a previously published Riemannian method, Riemannian Procrustes Analysis (RPA). Interestingly, tangent space alignment has an intrinsic ability to deal with transfer learning for sets of data that have different number of channels, naturally applying to inter-dataset transfer learning.

KEYWORDS

Brain-Computer Interface, Riemannian geometry, transfer learning, domain adaptation, ERP, motor imagery, SSVEP

1. Introduction

A Brain-Computer Interface (BCI) is a system that allows interactions between a human and a machine using only neurophysiological signals coming from the brain. It aims at rehabilitating, improving, or enhancing the ability of the user by means of a computerized system (Wolpaw et al., 2002). The most common modality used to record neurophysiological signals is electroencephalography (EEG). This is mainly because EEG is affordable, completely safe for the user and because it features a high temporal resolution. EEG signals can be translated into a command to be sent to a computer by means of a decoding algorithm. The loop is often closed by means of a feedback given to the user.

Several BCI applications have emerged to help patients, such as spellers (Yin et al., 2015; Rezeika et al., 2018) or wheelchair controllers (Li et al., 2013). The focus in this line of research is to restore lost communication or movement capabilities. Other applications are designed for the rehabilitation of patients after an incapacitating event such as a stroke (Frisoli et al., 2012; Mane et al., 2020). Non-clinical applications have also been proposed, for example to provide a means of control in video games (Congedo et al., 2011; Bonnet et al., 2013). Mixed approaches are also possible, for example in the Cybathlon BCI Race, where people with complete or severe loss of motor function compete in a video game-based competition (Perdikis et al., 2018).

Several paradigms can be used in order to control a BCI. The most common are event related potentials (ERP), motor imagery (MI), and steady state visually evoked potentials (SSVEP). The ERP paradigm consists of electrical potentials evoked by sensory stimulations; in MI the user imagines to move body parts, resulting in synchronizations and desynchronizations in the sensory-motor cortex; in SSVEP-based BCIs, the user concentrates on visual stimuli flashed at distinct frequencies, leading to responses at the same frequency in the brain. Regardless of the paradigm, it is necessary to calibrate the BCI system in order to allow proper decoding. The calibration process is time consuming, annoying for the healthy user and problematic for the clinical population, which has limited mental resources (Mayaud et al., 2016). In fact, a calibration is required not only for every new user, but also for every new session of the same user. This is due to the high inter-subject and inter-session variability of the features extracted from the EEG. Such variability is caused by several factors, including, but not limited to, the impedance and placement of EEG electrodes, individual morphological and physiological characteristics of the brain and changing brain states.

One way to deal with this variability is to use transfer learning (TL). This means trying to reuse some of the information we have already gathered on known data that may be coming from either previous subjects or previous sessions. In transfer learning we usually consider two types of data. The **source** represents the data we already know on a given subject whereas the **target** consists of a new subject whose some training data may be available, but mostly is unlabeled and is to be used as a test. The aim is to adapt as accurately as possible the data of the target using the few available training data to the source data (or vice versa). In order to do so, several methods have already been developed.

The authors in Jayaram et al. (2016) adapt the weights given to spatial features that are meant to predict the stimulus in order to transfer information from one subject to another or from one session to another. Some other methods adapt the parameters of a neural network. For example the authors in Fahimi et al. (2019) perform a partial retraining of a deep neural network on a small number of samples of a new user, improving significantly the accuracy. Unsupervised domain adaptation methods have

also emerged, as in Sun et al. (2015), where the authors perform unsupervised transfer learning in the Euclidean domain, using covariance matrices to align data from different subjects. A well-established approach for classification in the BCI field is to use covariance matrices of the signal since those matrices have many relevant properties (Congedo et al., 2017). The covariance matrices are Symmetric Positive Definite (SPD) and therefore lie in a Riemannian manifold. In this way, some algorithms have been developed to achieve transfer learning in the Riemannian manifold of SPD matrices. For instance, the authors in Zanini et al. (2018) propose a recentering procedure consisting in translating the center of mass of both the source and target data to the identity using parallel transport. This procedure is actually equivalent to a whitening using the Riemannian mean as anchor point. In Yair et al. (2019), both the center of mass of the source and target data are translated to their midpoint along the geodesic, allowing equivalent results. The authors of Rodrigues et al. (2019), inspired by the Procrustes analysis, proposed to add two more steps after recentering: a stretching of the observations, so as to equalize the dispersion of the data in the source and target domain and a rotation, so as to align as much as possible the center of mass of each class between the source and the target data set. The method, named Riemannian Procrustes Analysis (RPA), was shown to allow efficient transfer learning. A later alignment method was discussed in He and Wu (2020). This method is similar to Sun et al. (2015) with improvement related to enhanced dimensions in the Euclidean space. The authors of Zhang et al. (2020) chose another approach by transferring instances of the source close enough to the target in order to enhance the low data availability of the target model. They used MI data and compared the proximity of source and target trials using Hamming distance after preprocessing steps. Another idea proposed in Zhang and Wu (2020) is to find a common subspace between source and target, yielding a projection matrix to reduce the gap between the source and the target. Finally the authors train on the source subspace to test on target subspace.

In this article we introduce a Riemannian transfer learning approach similar in spirit to the RPA approach (Rodrigues et al., 2019), but operating in the tangent space. Our contribution has multiple benefits as compared to previous attempts. First, it lies in a state-of-the-art BCI feature space, the Riemannian tangent space, introduced in the BCI domain by Barachant et al. (2010). Since the tangent space is an Euclidean space, there exists a wide variety of well-established tools to decode the data therein and in general they are faster as compared to a decoding approach in the Riemannian manifold. Second, since it acts on an Euclidean space, it can be used for all kind of feature vectors, not just those obtained in a Riemannian setting. Third, our method is computationally effective, as it only requires one singular value decomposition (SVD). Fourth, it extends naturally to the heterogeneous transfer learning case, i.e., when the number and/or placements of electrodes is not the same in the source

and target data set. In a similar previous attempt the SVD has been applied independently on the source and target dataset and the resulting matrices are then used to align the data (Sun et al., 2015). Our method is instead casted as a Procrustes problem and therefore it fulfills a well-known optimality condition for inter-domain alignment.

A previous version of our method has been presented in Bleuze et al. (2021). As compared to that presentation, we have improved it by adding several ways to deal with the rank deficiency of the cross product matrix. Also, here we test it on a very large amount of data, namely, 18 BCI databases comprising a total of 349 subjects. Furthermore, these databases pertain to three BCI modalities: event related potentials (ERP), motor imagery (MI), and steady state visually evoked potentials (SSVEP). Therefore, the present study is a comprehensive test bed, which is a well-grounded way to reach general conclusions when comparing machine learning pipelines.

2. Materials and methods

2.1. Notations

Throughout this article we will denote matrices with upper case bold characters (\mathbf{A}), vectors with lower case bold characters (\mathbf{a}), indices and scalars by lower case italic characters (a), and constants by upper case italics (A). The function $tr(\cdot)$ will indicate the trace of a matrix, $(\cdot)^T$ its transpose, $\|\cdot\|$ the 2-norm or the Frobenius norm, \circ the Hadamard product, $\log(\cdot)$, and $\exp(\cdot)$ the matrix logarithm and exponential, respectively. \mathbf{I}_N will denote the identity matrix in dimension N .

2.2. Riemannian geometry

Let us consider a set of trials $\{\mathbf{X}_n\}_{n \in [1, N]}$ with shape (N_c, N_{sample}) , where N_c is the number of channels, N_{sample} the number of (temporal) samples and N the number of matrices in the set. A generic trial is simply denoted as \mathbf{X} . In order to be as close as possible to a realistic scenario, we consider data with a low level of pre-processing and we do not use any artifact removal method, such as ocular artifacts or outliers removal (Çinar and Acir, 2017; Minguillon et al., 2017).

The (spatial) sample covariance matrix estimation (SCM) writes

$$\mathbf{C} = \frac{1}{N_{sample} - 1} \mathbf{X}\mathbf{X}^T. \tag{1}$$

The SCM has shape (N_c, N_c) . It lies in a Riemannian manifold of symmetric positive definite (SPD) matrices (Bhatia, 2009). It is therefore possible to classify directly a set $\{\mathbf{C}_n\}$ of covariance matrices by means of classification algorithms acting on such a manifold, such as the Minimum Distance to Riemannian Mean (MDRM) classifier (Barachant et al., 2012) or its refinement

Riemannian Minimum Distance to Means Field (RMDMF; Congedo et al., 2019). It is also possible to project the matrices onto the tangent space of the manifold at a base point \mathbf{M} and use Euclidean classifiers therein (Barachant et al., 2012, 2013). The base point \mathbf{M} in this work will always be chosen as the Log-Euclidean mean, which is defined as Fillard et al. (2005).

$$\mathbf{M} = \exp\left(\frac{1}{N} \sum_n \log(\mathbf{C}_n)\right). \tag{2}$$

The projection onto the tangent space at base point \mathbf{M} is obtained by the logarithmic map operator (Nielsen and Bhatia, 2013).

$$\text{Log}_{\mathbf{M}}(\mathbf{C}) = \mathbf{M}^{\frac{1}{2}} \log\left(\mathbf{M}^{-\frac{1}{2}} \mathbf{C} \mathbf{M}^{-\frac{1}{2}}\right) \mathbf{M}^{\frac{1}{2}}. \tag{3}$$

The projected matrix is now a (N_c, N_c) symmetric matrix. Since we are concerned with transfer learning (TL), we are interested in matching the position of the source and target data sets in the manifold as much as possible. Following Zanini et al. (2018), we recenter both the source and target data sets by setting their global mean at the identity. This is simply obtained by transforming all trials of a dataset such as

$$\mathbf{C}_{\text{rec}} = \mathbf{M}^{-\frac{1}{2}} \mathbf{C} \mathbf{M}^{-\frac{1}{2}}, \tag{4}$$

where \mathbf{M} is the center of mass of the observations and \mathbf{C}_{rec} denotes the recentered trial. After recentering all trials their center of mass becomes the identity matrix, corresponding to the “zero” point in an Euclidean space.

The logarithmic mapping at the identity simplifies, yielding

$$\begin{aligned} \text{Log}_{\mathbf{I}_{N_c}}(\mathbf{C}_{\text{rec}}) &= \mathbf{I}_{N_c}^{1/2} \log\left(\mathbf{I}_{N_c}^{-1/2} \mathbf{M}^{-\frac{1}{2}} \mathbf{C} \mathbf{M}^{-\frac{1}{2}} \mathbf{I}_{N_c}^{-1/2}\right) \mathbf{I}_{N_c}^{1/2} \\ &= \log\left(\mathbf{M}^{-\frac{1}{2}} \mathbf{C} \mathbf{M}^{-\frac{1}{2}}\right). \end{aligned}$$

The above recentering followed by tangent space projection was first proposed in the BCI field in Barachant et al. (2012, 2013) and is nowadays a standard processing procedure, which in this article is carried out systematically, unless explicitly mentioned.

Once projected onto the tangent space the matrices are vectorized. Since they are symmetric, only the upper (or lower) triangle of the matrix is kept and the off-diagonal terms are weighted by $\sqrt{2}$ so as to preserve the norm of the original matrix. In mathematical notation, the vectorization of tangent vector \mathbf{S} reads

$$\mathbf{s} = \text{triu}(\mathbf{S} \circ \mathbf{A}), \tag{5}$$

with $\text{triu}(\cdot)$ the operator vectorizing the upper triangle and \mathbf{A} a matrix with the same shape as \mathbf{S} , filled with 1 on the diagonal and $\sqrt{2}$ on the off-diagonal part. Since the matrices have been previously recentered, the resulting vectors are also recentered, that is, the mean tangent vector is the zero vector.

Having obtained the tangent vectors as described here above, it is possible to use all the well know classification algorithms

that act in an Euclidean space, the most commonly employed in the BCI community being the linear discriminant analysis (LDA; Barachant et al., 2012), support vector classifier (SVC; Xu et al., 2021) and Lasso logistic regression (LR) (Tomioka et al., 2006). In this study, we use the SVC.

2.3. Alignment

As anticipated in the introduction, the method here proposed has been inspired by previous Riemannian TL methods, such as in Zanini et al. (2018) and Rodrigues et al. (2019) which focus on covariance matrices in the Riemannian space of SPD matrices. In Rodrigues et al. (2019) the authors consider again the recentering and add two further alignment steps:

1. a *rescaling* so as to match the dispersion around the mean (center of mass) in both the source and target data sets and
2. a *rotation* so as to align the mean of each class as much as possible. This effectively results in a Riemannian Procrustes alignment.

In this article, the same steps are undertaken in the tangent space. In particular, we rotate the tangent vectors using an Euclidean Procrustes procedure.

Let us consider the set of centered tangent vectors for the source $\{\mathbf{s}_n\}_{n \in [1, N_s]}$ and the target $\{\mathbf{t}_n\}_{n \in [1, N_t]}$ domain. N_s and N_t are the number of vectors for, respectively, the source and the target data set. As we will see, in the following it will not be required that the source and target tangent vectors have the same dimensions. Denoting \mathbf{s} and \mathbf{t} the generic source and target tangent vectors, the rescaling is obtained setting the average norm within each set to 1, which is readily obtained by transformations

$$\tilde{\mathbf{s}} = \frac{\mathbf{s}}{\frac{1}{N_s} \sum_n \|\mathbf{s}_n\|} \tag{6}$$

and

$$\tilde{\mathbf{t}} = \frac{\mathbf{t}}{\frac{1}{N_t} \sum_n \|\mathbf{t}_n\|}, \tag{7}$$

yielding rescaled source and target data sets $\{\tilde{\mathbf{s}}_n\}_{n \in [1, N_s]}$ and $\{\tilde{\mathbf{t}}_n\}_{n \in [1, N_t]}$. It is also possible to set the norm of the target data set equal to the norm of the source data set if it is sought not to modify the norm of the source data set.

For the rotation (alignment), we propose a supervised method that uses the mean point of the classes. Let us consider K classes that we ought to align. Although other procedures are possible, in the following we always align the target set to the source set. Let \mathbf{y} and \mathbf{z} be the label vectors of, respectively, $\{\tilde{\mathbf{s}}_n\}_{n \in [1, N_s]}$ and $\{\tilde{\mathbf{t}}_n\}_{n \in [1, N_t]}$ with shape N_s and N_t . We start by

computing the mean for each class k , given its N_k trials

$$\bar{\mathbf{s}}_k = \frac{1}{N_k} \sum_{y_i=k} \tilde{\mathbf{s}}_i \tag{8}$$

for the source set and

$$\bar{\mathbf{t}}_k = \frac{1}{N_k} \sum_{z_i=k} \tilde{\mathbf{t}}_i \tag{9}$$

for the target set. In the supervised procedure these vectors are the *anchor points* we use for alignment. Therefore, we define

$$\bar{\mathbf{S}} = [\bar{\mathbf{s}}_k, k \in [1, K]] \tag{10}$$

and

$$\bar{\mathbf{T}} = [\bar{\mathbf{t}}_k, k \in [1, K]] \tag{11}$$

as the two matrices of shape $(\frac{N_c(N_c+1)}{2}, K)$ holding the anchor vectors stacked one next to the other. We can now define the cross-product matrix

$$\mathbf{C}_{st} = \bar{\mathbf{S}} \bar{\mathbf{T}}^T. \tag{12}$$

of shape $(\frac{N_c(N_c+1)}{2}, \frac{N_c(N_c+1)}{2})$. Like any rectangular matrix—or squared when source and target have the same number of channels— \mathbf{C}_{st} can be decomposed by singular value decomposition, such that

$$\mathbf{C}_{st} = \mathbf{U} \mathbf{D} \mathbf{V}^T, \tag{13}$$

with \mathbf{U} and \mathbf{V} the two orthonormal matrices holding in columns the left and right singular vectors, respectively, and \mathbf{D} a diagonal matrix holding the singular values. As usual in signal processing, we will retain a subset of the singular vectors in order to suppress noise. Such a truncation has also the advantage to work for the case where \mathbf{U} and \mathbf{V} do not have the same shape. As a general rule, we seek the smallest number N_v of singular vectors which corresponding singular values explain at least 99.9% of the variance, resulting in $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ with shape $(\frac{N_c(N_c+1)}{2}, N_v)$. Finally, we are able to align the target vectors previously created $\{\tilde{\mathbf{t}}_n\}_{n \in [1, N]}$ to the domain of the source vectors $\{\tilde{\mathbf{s}}_n\}_{n \in [1, N]}$ as

$$\hat{\mathbf{t}} = \tilde{\mathbf{U}} \tilde{\mathbf{V}}^T \tilde{\mathbf{t}} \tag{14}$$

where $\hat{\mathbf{t}}$ denotes the aligned target vectors. The newly created set $\{\hat{\mathbf{t}}_n\}_{n \in [1, N]}$ is now aligned to the space of source vectors $\{\tilde{\mathbf{s}}_n\}_{n \in [1, N]}$, therefore it can be classified with algorithms trained on the source domain. As it is well-known, when the cross-product in Equation (12) is full-rank, the unique solution to the Procrustes optimization problem

$$\arg \min_{\mathbf{Z}} (\|\mathbf{Z} \bar{\mathbf{T}} - \bar{\mathbf{S}}\|) \tag{15}$$

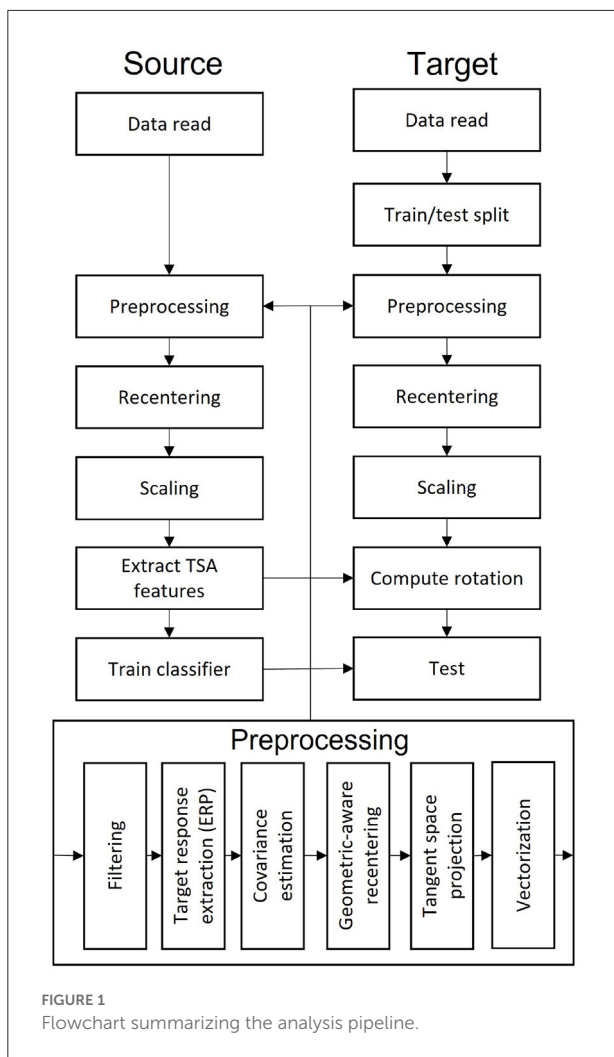


FIGURE 1
Flowchart summarizing the analysis pipeline.

is indeed $Z = UV^T$. In our case, the solution is not unique. Note that a connection between this problem and the Bures-Wasserstein metric has been recently described in Bhatia et al. (2019).

The whole process for our method is summarized in Figure 1.

2.4. Augmenting/improving C_{st}

Cross-product matrix C_{st} is usually rank deficient and its estimation could be improved in several ways. In this section, we will suggest two such improvements. First, as long as a supervised TL is possible, since we are relying on averaging the tangent vectors, it is possible to employ robust average estimators. For instance, we may consider the trimmed means,

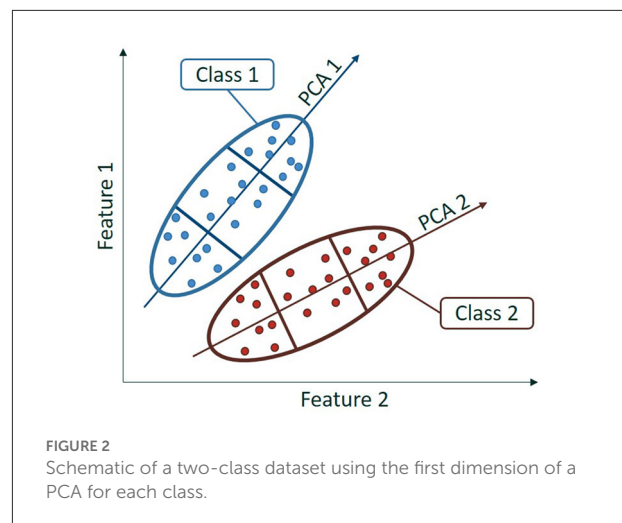


FIGURE 2
Schematic of a two-class dataset using the first dimension of a PCA for each class.

the median or power means to estimate suitable anchor points. We may also stack several of these average estimators to obtain larger matrices \bar{s}_k and \bar{t}_k , which may provide more robust information on the actual central tendency of the data.

Second, we cluster the data sets in several subsets describing the shape of the data set when considered altogether and compute separate means for each cluster. We may use, for instance, principal component analysis (PCA) on each class independently to create clusters, as depicted in Figure 2. The centroid of those clusters are then computed and used as anchor points. In order to obtain the clustering for both the source and target data set, we consider for each class a PCA trained on the source and used as such on both the source and target data set.

Using such a clustering procedure, if the source and the target data set display a rather similar shape, their alignment will be very effective, leading to promising transfer learning results. Such a procedure is also possible with unlabeled data in case of unsupervised TL. However, in this case, we have noticed that at least two PCA components are necessary to obtain an efficient transfer learning. Therefore, in the unsupervised case we recommend using at least two PCA components and separate data for each dimensions, creating a C_{st} matrix with shape $\left[\frac{N_c(N_c+1)}{2}, N_d \times N_g \right]$ with N_d the number of dimensions used and N_g the number of groups created in each dimensions. An effective strategy is to visualize the data and their representations in order to verify whether the chosen reduced dimensionality offers a good approximation of the data as it may be as well totally inaccurate, depending on the data, especially for unsupervised TL. For our results, we chose to create three PCA clusters for each class and use these means to compute the cross-product matrix C_{st} as it gives enough information without reducing the size of the data used for each mean too much.

TABLE 1 List of the processed databases for event-related potentials, motor imagery paradigms, and steady-state visual evoked potentials.

	Type	Ch	Trials	Sess	Sub
2003–2015	P300	8	1,800	1	10
2008–2014	P300	8	4,200	1	8
2009–2014	P300	16	1,728	3	10
Brain Invaders 2013a	P300	16	480	1	24
EPFL P300 dataset	P300	32	3,268	4	8
2001–2014	MI	22	288	2	9
2001–2015	MI	13	400	2	12
2002–2014	MI	15	160	1	14
2004–2014	MI	3	720	5	9
Alexandre motor imagery	MI	16	40	1	8
Cho 2017	MI	64	200	1	49
Grosse-Wentrup 2009	MI	128	300	1	10
Physionet motor imagery	MI	64	45	1	109
Weibo 2014	MI	60	160	1	10
Zhou 2016	MI	14	290	3	4
SSVEP exoskeleton	SSVEP	8	160	1	12
SSVEP Nakanishi	SSVEP	8	180	1	9
SSVEP Wang	SSVEP	62	240	1	34

Ch, number of channels; Sess, number of sessions; Sub, number of subjects.

3. Results

The TSA algorithm previously introduced has been tested on three well know BCI paradigms: ERP, MI, and SSVEP. We have analyzed 18 open-access BCI databases available on the Mother Of All BCI Benchmark (MOABB; Jayaram and Barachant, 2018). Python library, of which five uses the ERP paradigm, 10 the MI paradigm, and 3 the SSVEP paradigm. The 18 databases include a total of 349 subjects with very high variability between and within datasets. We summarize the data in Table 1, following Congedo et al. (2019).

We execute transfer learning from one subject to the other for all possible source/target pair of subjects within each database. The accuracy is evaluated using balanced accuracy since the number of trials per class is often unbalanced (always for the ERP paradigm).

Since the amount of data of all pair-wise comparisons is huge, we start by visually evaluating all balanced accuracies obtained for a given database by means of seriation plots, i.e., plots showing the accuracy for each pair of target and source subject arranged in matrix form. The accuracy is averaged over all numbers of alignment trials for each pair. The case where the target and the source are the same, i.e., on the diagonal, is replaced by the classical train-test cross-validation accuracy, offering a straightforward benchmark. Furthermore, the target and source subjects are sorted on rows by descending order of the train-test cross-validation accuracy. It should be kept

in mind that since the train-test procedure is fully supervised and optimized for the train data, it is expected to outperform a transfer learning method. Figure 3 shows a representative seriation plot for each paradigm allowing the visual comparison of the performance of the TSA vs. the RPA transfer learning method; all figures are available as Supplementary material.

In order to evaluate the average performance, we plot the balanced accuracy averaged across all subjects in a database for each method as a function of the number of alignment trials. Since we are averaging across subjects, for this analysis we include only those subjects featuring at least 60% accuracy in a classical train-test cross-validation. This restriction excludes about half of the subjects, leaving 178 subject out of 349. It's important to note that subjects with an overall 60% accuracy usually have more than 70% accuracy with all available training trials. Figure 4 shows a representative plot for each paradigm. All figures are available as Supplementary material.

Then, we summarize all the pair-wise accuracy information in the accuracy tables such as Table 2. These tables give for each target subject the accuracy averaged over numbers of alignment trials and source subjects. For this database, there are 30 numbers of alignment trials considered and 18 possible source subjects, which makes an average over 540 values. This makes the standard error low in general. Accuracy tables for all databases are given as Supplementary material.

The accuracy tables confirm what can be evaluated visually in the average accuracy plots and seriation plots; on the average there is about 1% difference between classical train-test cross-validation accuracy and TSA and about 5% between classical train-test cross-validation and RPA. This speaks in favor of a clear improvement of the TSA method over the RPA method. Table 3 summarizes all the balanced accuracy for each dataset. On the average across databases there is no loss of accuracy using a TSA as compared to the optimal train-test accuracy. This is not true for the RPA.

Finally, we performed statistical tests on all pair-wise source/target accuracy results we have collected. To this end, we follow the procedure introduced in Rodrigues et al. (2021). In a nutshell: we first compute signed paired *t*-test for every target subject comparing the accuracy between methods, yielding *T* statistics $T_{m,i}$ and *p*-values $p_{m,i}$ for each pair of methods *m* and target subject *i*. In order to correct for the multiplicity of statistical tests we use Holm's sequential rejection multiple test procedure (Holm, 1979) for each target subject. This produces tables such as Table 4. Corresponding tables for each database are available as Supplementary material. Then, we combine the *p*-values we obtain using the Stouffer's Z-score method (Zaykin, 2011) for each database, yielding multiple *p*-values corresponding to each pair of methods for each database. Those *p*-values are also corrected by means of Holm's procedure and are summarized in Table 5. In this table we can see that among the 18 databases we have analyzed, 11 show a significant improvement of TSA as compared to RPA.

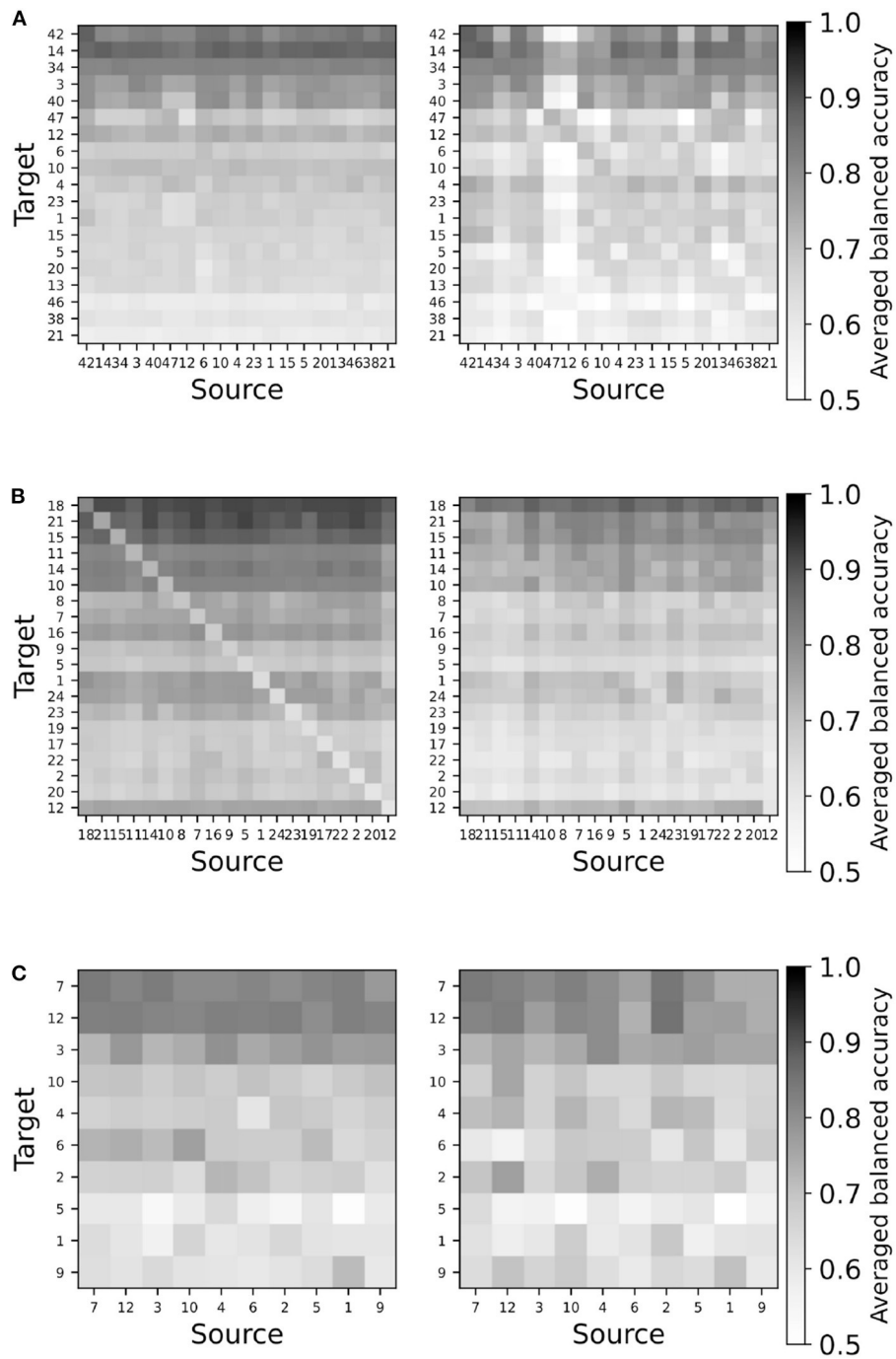
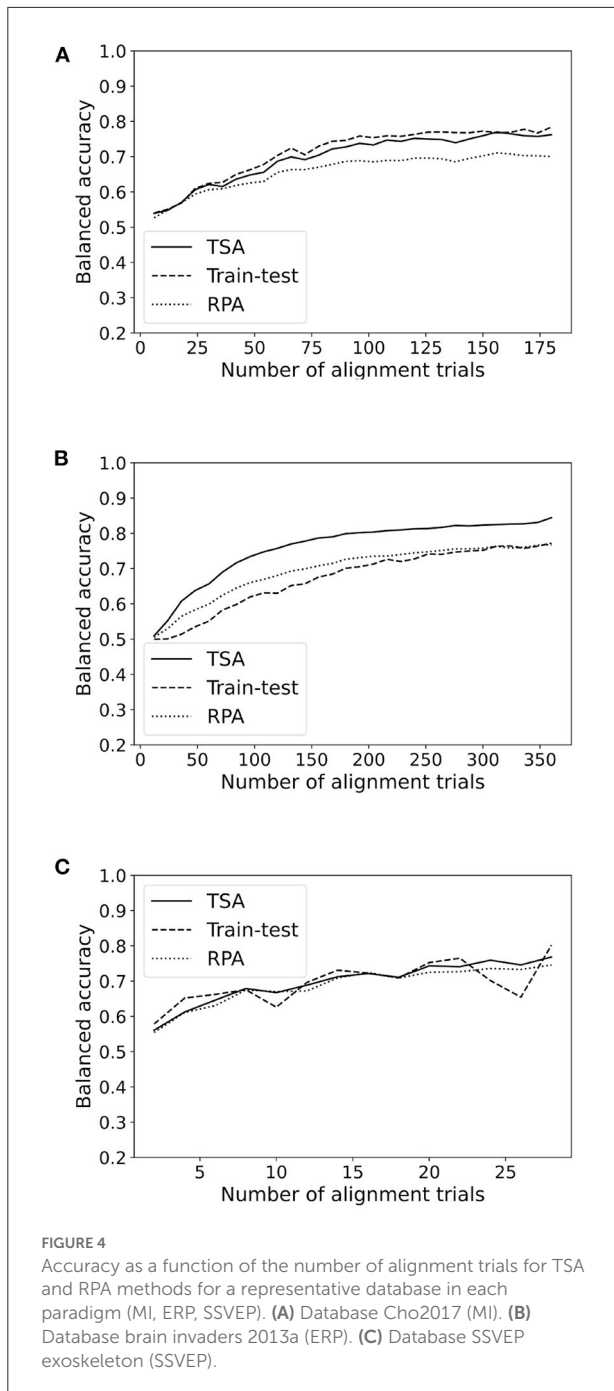


FIGURE 3
 Representative seriation plots for TSA (left) and RPA (right) methods for each paradigms. See text for details. **(A)** Database Cho2017 (MI). **(B)** Database brain invaders 2013a (ERP). **(C)** Database SSVEP exoskeleton (SSVEP).

No significant difference between the classical train-test cross-validation accuracy and TSA is found with the exception of two databases, for which TSA proves inferior. This number

grows to five databases comparing the classical train-test cross-validation accuracy and the accuracy obtained by RPA. Finally, using Stouffer’s Z-score method, *p*-values corresponding to each



paradigm are computed and corrected with Holm’s procedure (Table 6).

So far we have focused on cross-subject transfer learning, however the method we propose can also be used to transfer different information, such as from one session to another with the same subject or from one task to the another. In order to ensure the ability of our method to reduce the inter-session variability, we used the dataset having multiple sessions and

TABLE 2 Balanced accuracy ± standard error for the good subjects of database Cho 2017 averaged over number of alignment trials and source for train-test, TSA, and RPA.

	Train-test acc	TSA acc	RPA acc
Subject 1	67.86 ± 2.00	66.70 ± 0.43	65.84 ± 0.43
Subject 3	81.39 ± 0.81	77.78 ± 0.18	74.96 ± 0.32
Subject 4	69.26 ± 1.49	69.24 ± 0.40	69.25 ± 0.50
Subject 5	65.90 ± 2.12	65.03 ± 0.50	59.12 ± 0.55
Subject 6	70.74 ± 2.11	67.65 ± 0.38	59.80 ± 0.53
Subject 10	70.55 ± 1.91	70.71 ± 0.45	62.32 ± 0.43
Subject 12	70.97 ± 1.61	73.01 ± 0.39	67.68 pm0.29
Subject 13	63.60 ± 1.14	64.30 ± 0.25	60.83 ± 0.25
Subject 14	88.26 ± 1.52	86.99 ± 0.37	82.06 ± 0.45
Subject 15	66.15 ± 1.64	65.63 ± 0.38	65.20 ± 0.37
Subject 20	64.96 ± 1.13	64.64 ± 0.26	60.79 ± 0.33
Subject 21	60.63 ± 1.50	58.63 ± 0.24	56.54 ± 0.30
Subject 23	68.64 ± 1.62	66.69 ± 0.37	64.94 ± 0.41
Subject 34	82.73 ± 0.37	81.89 ± 0.09	79.97 ± 0.15
Subject 38	61.98 ± 1.27	61.51 ± 0.27	58.53 ± 0.35
Subject 40	76.73 ± 1.89	76.30 ± 0.44	72.21 ± 0.47
Subject 42	88.34 ± 3.32	83.02 ± 0.73	75.78 ± 0.77
Subject 46	62.55 ± 1.79	59.08 ± 0.37	54.64 ± 0.26
Subject 47	72.34 ± 1.66	67.98 ± 0.36	63.03 ± 0.42

TABLE 3 Balanced accuracy ± standard error for the good subjects of each dataset averaged over number of alignment trials, target and source for train-test, TSA, and RPA.

	Train-test acc	TSA acc	RPA acc
2003–2015	69.04 ± 0.74	76.05 ± 0.57	69.63 ± 0.57
2008–2014	67.52 ± 0.69	72.00 ± 0.38	65.50 ± 0.38
2009–2014	70.48 ± 0.50	77.38 ± 0.21	70.90 ± 0.21
Brain invaders 2013a	67.26 ± 0.45	76.07 ± 0.11	69.59 ± 0.11
EPFL P300 dataset	67.22 ± 0.61	73.013 ± 0.24	65.28 ± 0.24
2001–2015	81.13 ± 0.73	79.01 ± 0.24	76.21 ± 0.24
2002–2014	78.40 ± 0.70	76.51 ± 0.20	72.55 ± 0.55
2004–2014	74.35 ± 1.30	76.00 ± 0.59	75.27 ± 0.59
Alexandre motor imagery	78.51 ± 2.82	76.30 ± 1.33	75.93 ± 1.33
Cho 2017	71.24 ± 0.52	69.83 ± 0.12	65.97 ± 0.12
Grosse-Wentrup 2009	77.20 ± 0.91	73.12 ± 0.34	72.16 ± 0.34
Physionet motor imagery	70.75 ± 1.35	64.28 ± 0.27	63.26 ± 0.27
Weibo 2014	73.97 ± 1.25	69.66 ± 0.50	71.27 ± 0.50
Zhou 2016	82.82 ± 1.59	82.43 ± 0.86	81.31 ± 0.86
SSVEP exoskeleton	69.50 ± 1.16	69.69 ± 0.40	68.72 ± 0.40
SSVEP Nakanishi	95.55 ± 0.82	95.95 ± 0.28	96.83 ± 0.28
SSVEP Wang	68.26 ± 2.63	57.42 ± 0.76	59.11 ± 0.76
Global	74.42 ± 0.30	74.49 ± 0.12	71.79 ± 0.12

processed results for inter-session cross-validation. In order to do so, we used the data of one session as a source, then for each other sessions with 80% test and 20% training data split we trained the transfer learning model and tested the results. The

TABLE 4 Subject-wise *p*-values for MI database Cho 2017.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
Subject 1	0.002*	0.818	0.420
Subject 3	0.033*	<0.001*	<0.001*
Subject 4	0.005*	0.690	0.248
Subject 5	<0.001*	0.980	0.301
Subject 6	0.001*	0.379	0.013*
Subject 10	<0.001*	0.503	0.007*
Subject 12	<0.001	0.086	<0.001*
Subject 13	0.001*	0.997	0.700
Subject 14	0.443	<0.001*	<0.001*
Subject 15	0.004*	0.964	0.843
Subject 20	0.385	0.991	0.988
Subject 21	<0.001*	1.000	0.999
Subject 23	0.009*	0.836	0.514
Subject 34	<0.001*	<0.001*	<0.001*
Subject 38	<0.001*	1.000	0.980
Subject 40	0.020*	0.002*	<0.001*
Subject 42	0.361	<0.001*	<0.001*
Subject 46	0.126	1.000	0.995
Subject 47	<0.001*	0.038*	<0.001*

TRAIN, train-test; *Significant *p*-values after multiple comparison correction.

TABLE 5 Database-wise *p*-values.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
2003–2015	< 0.001	1.000	0.663
2008–2014	0.002*	0.877	0.326
2009–2014	<0.001*	1.000	0.561
Brain invaders 2013a	<0.001*	1.000	1.000
EPFL P300 dataset	<0.001*	1.000	0.102
2001–2014	0.004*	0.468	0.191
2001–2015	<0.001*	0.133	0.005*
2002–2014	<0.001*	0.139	<0.001*
2004–2014	0.515	0.641	0.552
Alexandre motor imagery	0.636	0.353	0.280
Cho 2017	<0.001*	0.098	<0.001*
Grosse-Wentrup 2009	0.097	0.083	0.071
Physionet motor imagery	<0.001*	<0.001*	<0.001*
Weibo 2014	1.000	0.099	0.189
Zhou 2016	0.071	0.453	0.398
SSVEP exoskeleton	0.002*	0.702	0.477
SSVEP Nakanishi	1.000	0.464	0.886
SSVEP Wang	0.989	<0.001*	<0.001*

TRAIN, train-test; *Significant *p*-values after multiple comparison correction.

processed is then repeated using each session as the source. We compared four different methods:

- Tangent Space Alignment (TSA), our method,
- Riemannian Procrustes Analysis (RPA) used as a comparison in this article,

TABLE 6 *p*-values for each paradigm and in global for all tests that have been done. The global *p*-values are the combination of the *p*-values for all databases regardless of their paradigm.

	TSA>RPA	TRAIN>TSA	TRAIN>RPA
p300	<0.001*	1.000	0.961
Imagery	<0.001*	<0.001*	<0.001*
SSVEP	<0.654	<0.001*	<0.001*
Global	<0.001*	1.000	<0.001*

TRAIN, train-test; *Significant *p*-values after multiple comparison correction.

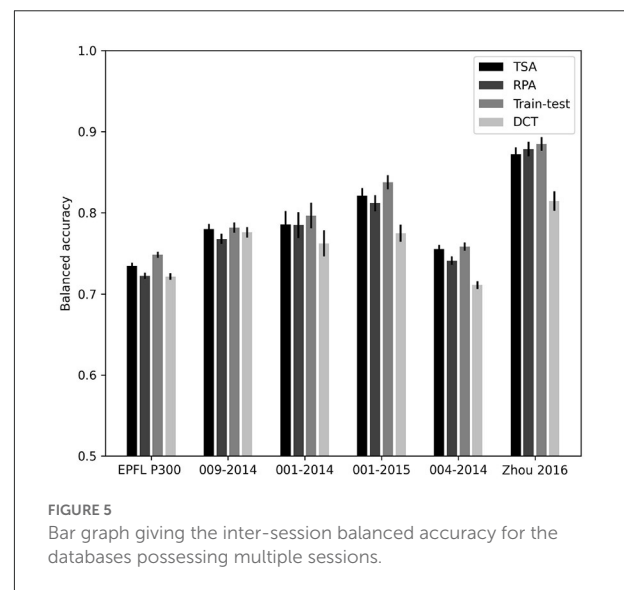


FIGURE 5 Bar graph giving the inter-session balanced accuracy for the databases possessing multiple sessions.

TABLE 7 Inter-session balanced accuracy table for each dataset and methods.

	Train-test	TSA	RPA	DCT
EPFL P300	74.85 ± 0.38	73.46 ± 0.38	72.24 ± 0.39	72.16 ± 0.40
2009–2014	78.19 ± 0.64	78.00 ± 0.65	76.79 ± 0.63	77.59 ± 0.64
2001–2014	79.66 ± 1.57	78.56 ± 1.66	78.48 ± 1.61	76.23 ± 1.63
2001–2015	83.78 ± 0.86	82.10 ± 0.95	81.19 ± 0.99	77.49 ± 1.03
2004–2014	75.83 ± 0.52	75.55 ± 0.50	74.11 ± 0.53	71.09 ± 0.50
Zhou 2016	88.48 ± 0.86	87.22 ± 0.84	87.83 ± 0.90	81.45 ± 1.19

- Usual train-test method using only target data,
- Direct testing (DCT) using algorithms trained on the source without aligning target data by a rotation (recentering only).

The results are given in Figure 5 and presented numerically in Table 7.

These results are coherent with the previous ones. They yield accuracies slightly improved as compared to sessions mixed all together, which is expected. Moreover, the proposed method

performs better than RPA on all databases with the exception of Zhou 2016.

4. Discussion

The extensive analysis we have carried out shows that for the ERP paradigm TSA clearly outperforms RPA. For the MI paradigm we observe that TSA performs better than RPA and that the classical train-test cross-validation outperforms both TSA and RPA. For SSVEP, the classical train-test cross-validation outperforms both TSA and RPA. In global, RPA is outperformed by both TSA and the classical train-test cross-validation. Based on Table 6, we do not conclude on the superiority of TSA as compared to train-test in terms of accuracy as the global value that can be found is mainly due to very large values found for the ERP paradigm. To conclude with the results, it has been shown that TSA outperforms RPA for the large majority of the databases we have used in this analysis, reaching an accuracy pretty close to the optimal train-test method. It is also a clear improvement from a methodological point of view in comparison to RPA as it naturally allows transfer learning between datasets with different number of channels, where RPA needs some extensions in order to do so (Rodrigues et al., 2021).

In this paper, we have introduced a new method for transfer learning inter and intra subject for brain computer interfaces. Our study indicates that it outperforms a state-of-the-art analogous method (RPA). However, it still does not reach the same accuracy that can be achieved with a classical test-train cross-validation procedure for the motor imagery and SSVEP paradigm. Further research is needed to understand why the performance of the TSA method is clearly superior for the ERP paradigm.

In terms of computation time, since we have a closed form for the rotation of TSA method, it is way faster than the RPA method, where an optimization on the Grassmann manifold is performed. However, even if our method can be faster by one order of magnitude, with N_c being the size of the pre-covariance signal (number of channels plus number of channels of the average target for ERP, number of channels for MI, number of classes times number of channels for SSVEP) we compute rotations with size $[\frac{N_c*(N_c+1)}{2}, \frac{N_c*(N_c+1)}{2}]$ where RPA computes rotation with size (N_c, N_c) . This means that for datasets with a significant amount of channels the computational advantage of TSA will tend to vanish. Table 8 shows the average time of computation for each dataset if no spatial filter were to be applied before the covariance estimation, sometimes yielding big covariance matrices. Usually spatial filter are applied so the low time values are those that are to be usually encountered. Moreover, when computing the rotations for TSA, we can consider only a number of channels that will result in a proportion of

TABLE 8 Database-wise average computation time for both TSA and RPA methods.

	TSA (s)	RPA (s)	Ratio	Cov shape
2003–2015	1.027	11.889	11.57	16 * 16
2008–2014	1.711	10.874	6.36	16 * 16
2009–2014	0.897	6.118	6.82	32 * 32
Brain invaders 2013a	0.774	13.368	17.26	32 * 32
EPFL P300 dataset	1.341	8.607	6.42	64 * 64
2001–2014	0.999	5.075	5.08	22 * 22
2001–2015	0.664	4.362	6.57	13 * 13
2002–2014	0.416	2.241	5.39	15 * 15
2004–2014	0.029	0.166	5.74	3 * 3
Alexandre motor imagery	0.067	0.777	11.61	16 * 16
Cho 2017	5.299	16.858	3.18	64 * 64
Grosse-Wentrup 2009	161.942	113.926	0.70	128 * 128
Physionet motor imagery	2.739	4.432	1.62	64 * 64
Weibo 2014	3.719	13.282	3.57	60 * 60
Zhou 2016	0.468	3.752	8.02	14 * 14
SSVEP exoskeleton	0.056	1.184	21.22	16 * 16
SSVEP Nakanishi	0.41	0.970	23.81	16 * 16
SSVEP Wang	125.949	22.320	0.18	124 * 124

the global variance of the data. This means that we do not need to compute the whole $[\frac{N_c*(N_c+1)}{2}, \frac{N_c*(N_c+1)}{2}]$ rotation matrix, we can select only a few singular values and their corresponding vectors, highly reducing computation time if needed.

We also observed that in some cases, mainly for ERPs, skipping the rescaling of the target data will lead to improved classification results. Further research is needed to fully understand the role of rescaling in Procrustes-like transfer learning methods.

In this article, we have proposed a procedure to improve the rank deficiency of the cross-product matrix, making the result more stable and more accurate. However, this improvement also presents some downsides. When using only the average point of the data for both source and target data, the method will be pretty sensitive to noise since the number of points used for means computation is drastically reduced. Adding trimmed means and/or medians could make the method less sensitive to noise. Additionally, when augmenting C_{st} using PCA, the more groups will be created, the more each group will be sensitive to noise. By reducing the number of points in a group one increases the impact of artifacts on the average point. Furthermore, PCA is an algorithm that is pretty sensitive to noise and could be replaced by one of its robust variant. It is to be noticed that a low number of groups in general allows a good approximation of the shape of the data. Of course, artifact correction or removal would allow better performances.

It should be reminded that in this article we did not apply any form of pre-processing in order to increase the overall accuracy. We have done so because the goal of the article is to compare transfer learning methods on a large amount of data and in a wide variety of real-world situations, that is, on noisy data. For the same reason, we are using all the data of a subject without making any adaptation from one session to the other. This choice obviously leads to reduced overall accuracy, resulting in an important decrease in the number of subjects used for the final average results. It is important to do so, however, since, as it has been found in previous studies, there is a large gap between “good” and “bad” subjects in transfer learning accuracy (Zanini et al., 2018; Rodrigues et al., 2019).

Another point to mention is that even though our method has been tested extensively on many databases, there are even more databases to test on. Additionally, some paradigms such as affective BCI have not been investigated in this article. Investigation on cross-database transfer learning is still to be done.

As all efficient transfer learning methods, TSA can be very helpful when used along with a machine learning model that takes too much time for training on new data for online sessions. TSA also allows the alignment of multiple subjects' data into the same feature space. Such alignment could improve classification accuracy of multiple subjects and allow the training of robust classifier on aligned data that will give improved results for new subjects once they are aligned. This is the object of current investigation in our laboratory.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found at: <http://moabb.neurotechx.com/docs/datasets.html>.

References

- Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. (2012). Multiclass brain-computer interface classification by Riemannian geometry. *IEEE Trans. Biomed. Eng.* 59, 920–928. doi: 10.1109/TBME.2011.2172210
- Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. (2013). Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing* 2013, 172–178. doi: 10.1016/j.neucom.2012.12.039
- Barachant, A., Bonnet, S., Congedo, M., Jutten, C., Riemannian, C. J., and Bonnet, S. (2010). Riemannian geometry applied to BCI classification. *Lecture Notes Comput. Sci.* 6365, 629–636. doi: 10.1007/978-3-642-15995-4_78
- Bhatia, R. (2009). *Positive Definite Matrices*. Princeton, NJ: Princeton University Press. doi: 10.1515/9781400827787
- Bhatia, R., Jain, T., and Lim, Y. (2019). On the Bures-Wasserstein distance between positive definite matrices. *Exposit. Math.* 37, 165–191. doi: 10.1016/j.exmath.2018.01.002
- Bleuze, A., Mattout, J., and Congedo, M. (2021). “Transfer learning for the Riemannian tangent space: applications to brain-computer interfaces,” in *7th International Conference on Engineering and Emerging Technologies, ICEET 2021 (Istanbul)*, 1–6. doi: 10.1109/ICEET53442.2021.9659607
- Bonnet, L., Lotte, F., and Lécuyer, A. (2013). Two brains, one game: design and evaluation of a multiuser BCI video game based on motor imagery. *IEEE Trans. Comput. Intell. AI Games* 5, 185–198. doi: 10.1109/TCIAIG.2012.2237173
- Çınar, S. and Acır, N. (2017). A novel system for automatic removal of ocular artefacts in EEG by using outlier detection methods and independent component analysis. *Expert Syst. Appl.* 68, 36–44. doi: 10.1016/j.eswa.2016.10.009
- Congedo, M., Barachant, A., and Bhatia, R. (2017). Riemannian geometry for eeg-based brain-computer interfaces; a primer and a review. *Brain Comput. Interfaces* 4, 155–174. doi: 10.1080/2326263X.2017.1297192
- Congedo, M., Goyat, M., Tarrin, N., Ionescu, G., Varnet, L., Rivet, B., et al. (2011). “Brain invaders”: a prototype of an open-source p300-based video

Author contributions

AB performed the research, analyzed the data, and wrote the manuscript. MC suggested the original idea for the method. MC and JM designed, reviewed, and edited the manuscript. All authors have read and approved the submitted manuscript.

Funding

This research has been partially supported by CNRS grant 80|PRIMES TrAp and by ANR grant Hifi (ANR-20-CE17-0023).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fnhum.2022.1049985/full#supplementary-material>

game working with the openvibe platform," in *5th International Brain-Computer Interface Conference, BCI 2011* (Graz), 280–283.

Congedo, M., Rodrigues, P. L. C., and Jutten, C. (2019). "The Riemannian minimum distance to means field classifier," in *8th International Brain-Computer Interface Conference, BCI 2019* (Graz).

Fahimi, F., Zhang, Z., Goh, W. B., Lee, T. S., Ang, K. K., and Guan, C. (2019). Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. *J. Neural Eng.* 16:026007. doi: 10.1088/1741-2552/aaf3f6

Fillard, P., Arsigny, V., Ayache, N., and Pennec, X. (2005). A Riemannian framework for the processing of tensor-valued images. *Lect. Notes Comput. Sci.* 3753, 112–123. doi: 10.1007/11577812_10

Frisoli, A., Loconsole, C., Leonardi, D., Banno, F., Barsotti, M., Chisari, C., et al. (2012). A new gaze-BCI-driven control of an upper limb exoskeleton for rehabilitation in real-world tasks. *IEEE Trans. Syst. Man Cybern. Part C* 42, 1169–1179. doi: 10.1109/TSMCC.2012.2226444

He, H., and Wu, D. (2020). Transfer learning for brain-computer interface: a Euclidean space data alignment approach. *IEEE Trans. Biomed. Eng.* 67, 1–22. doi: 10.1109/TBME.2019.2913914

Holm, S. (1979). Board of the foundation of the Scandinavian journal of statistics a simple sequentially rejective multiple test procedure a simple sequentially rejective multiple test procedure. *Scand. J. Stat.* 6, 65–70.

Jayaram, V., Alamgir, M., Altun, Y., Scholkopf, B., and Grosse-Wentrup, M. (2016). Transfer learning in brain-computer interfaces. *IEEE Comput. Intell. Mag.* 11, 20–31. doi: 10.1109/MCI.2015.2501545

Jayaram, V., and Barachant, A. (2018). MOABB: trustworthy algorithm benchmarking for BCIs. *J. Neural Eng.* 15:066011. doi: 10.1088/1741-2552/aadea0

Li, Y., Pan, J., Wang, F., and Yu, Z. (2013). A hybrid BCI system combining p300 and SSVEP and its application to wheelchair control. *IEEE Trans. Biomed. Eng.* 60, 3156–3166. doi: 10.1109/TBME.2013.2270283

Mane, R., Chouhan, T., and Guan, C. (2020). BCI for stroke rehabilitation: motor and beyond. *J. Neural Eng.* 17:041001. doi: 10.1088/1741-2552/aba162

Mayaud, L., Cabanilles, S., Langenhove, A. V., Congedo, M., Barachant, A., Pouplin, S., et al. (2016). Brain-computer interface for the communication of acute patients: a feasibility study and a randomized controlled trial comparing performance with healthy participants and a traditional assistive device. *Brain Comput. Interfaces* 3, 197–215. doi: 10.1080/2326263X.2016.1254403

Minguillon, J., Lopez-Gordo, M. A., and Pelayo, F. (2017). Trends in EEG-BCI for daily-life: requirements for artifact removal. *Biomed. Signal Process. Control* 31, 407–418. doi: 10.1016/j.bspc.2016.09.005

Nielsen, F., and Bhatia, R. (2013). *Matrix Information Geometry*. Heidelberg. doi: 10.1007/978-3-642-30232-9

Perdikis, S., Tonin, L., Saeedi, S., Schneider, C., and del R. Millán, J. (2018). The cyathlon bci race: Successful longitudinal mutual learning

with two tetraplegic users. *PLoS Biol.* 16:5:e2003787. doi: 10.1371/journal.pbio.2003787

Rezeika, A., Benda, M., Stawicki, P., Gemblar, F., Saboor, A., and Volosyak, I. (2018). Brain-computer interface spellers: a review. *Brain Sci.* 8:57. doi: 10.3390/brainsci8040057

Rodrigues, P. L., Congedo, M., and Jutten, C. (2021). Dimensionality transcending: a method for merging BCI datasets with different dimensionalities. *IEEE Trans. Biomed. Eng.* 68, 673–684. doi: 10.1109/TBME.2020.3010854

Rodrigues, P. L. C., Jutten, C., and Congedo, M. (2019). Riemannian procrustes analysis: transfer learning for brain-computer interfaces. *IEEE Trans. Biomed. Eng.* 66, 2390–2401. doi: 10.1109/TBME.2018.2889705

Sun, B., Feng, J., and Saenko, K. (2015). Return of frustratingly easy domain adaptation. *Proc. AAAI Conf. Artif. Intell.* 30:1. doi: 10.1609/aaai.v30i1.10306

Tomioka, R., Aihara, K., and Müller, K.-R. (2006). Logistic regression for single trial EEG classification. *Adv. Neural Inform. Process. Syst.* 19, 1377–1384.

Wolpaw, J. R., Birbaumer, N., Mcfarland, D. J., Pfurtscheller, G., and Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clin. Neurophysiol.* 113, 767–791. doi: 10.1016/S1388-2457(02)00057-3

Xu, J., Markham, A., Meunier, A., Raggam, P., and Grosse-Wentrup, M. (2021). Distance covariance: a nonlinear extension of Riemannian geometry for EEG-based brain-computer interfacing. *IEEE Int. Conf. Syst. Man Cybern.* 2021, 2000–2005. doi: 10.1109/SMC52423.2021.9658876

Yair, O., Ben-Chen, M., and Talmon, R. (2019). Parallel transport on the cone manifold of spd matrices for domain adaptation. *IEEE Trans. Signal Process.* 67, 1797–1811. doi: 10.1109/TSP.2019.2894801

Yin, E., Zhou, Z., Jiang, J., Yu, Y., and Hu, D. (2015). A dynamically optimized ssvep brain-computer interface (BCI) speller. *IEEE Trans. Biomed. Eng.* 62, 1447–1456. doi: 10.1109/TBME.2014.2320948

Zanini, P., Congedo, M., Jutten, C., Said, S., and Berthoumieu, Y. (2018). Transfer learning: a Riemannian geometry framework with applications to brain-computer interfaces. *IEEE Trans. Biomed. Eng.* 65, 1107–1116. doi: 10.1109/TBME.2017.2742541

Zaykin, D. V. (2011). Optimally weighted z-test is a powerful method for combining probabilities in meta-analysis. *J. Evol. Biol.* 24, 1836–1841. doi: 10.1111/j.1420-9101.2011.02297.x

Zhang, K., Xu, G., Chen, L., Tian, P., Han, C. C., Zhang, S., et al. (2020). Instance transfer subject-dependent strategy for motor imagery signal classification using deep convolutional neural networks. *Comput. Math. Methods Med.* 2020:1683013. doi: 10.1155/2020/1683013

Zhang, W., and Wu, D. (2020). Manifold embedded knowledge transfer for brain-computer interfaces. *IEEE Trans. Neural Syst. Rehabil. Eng.* 28, 1117–1127. doi: 10.1109/TNSRE.2020.2985996