

Article

Transfer Learning Algorithm of P300-EEG Signal Based on XDAWN Spatial Filter and Riemannian Geometry Classifier

Feng Li ^{1,2,†}, Yi Xia ^{1,2,†}, Fei Wang ^{3,*} , Dengyong Zhang ^{1,2} , Xiaoyu Li ^{1,2} and Fan He ^{1,2}

¹ School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China; Lif@csust.edu.cn (F.L.); csustxy@163.com (Y.X.); zhdy@csust.edu.cn (D.Z.); csustecolee@163.com (X.L.); hf0208@stu.csust.edu.cn (F.H.)

² Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, Changsha University of Science and Technology, Changsha 410114, China

³ School of Software, South China Normal University, Guangzhou 510631, China

* Correspondence: scutauwf@foxmail.com

† These authors contributed equally to this work.

Received: 1 February 2020; Accepted: 2 March 2020; Published: 5 March 2020



Abstract: The electroencephalogram (EEG) signal in the brain–computer interface (BCI) has suffered great cross-subject variability. The BCI system needs to be retrained before each time it is used, which is a waste of resources and time. Thus, it is difficult to generalize a fixed classification method for all subjects. Therefore, the transfer learning method proposed in this article, which combines XDAWN spatial filter and Riemannian Geometry classifier (RGC), can achieve offline cross-subject transfer learning in the P300-speller paradigm. The XDAWN spatial filter is used to enhanced the P300 components in the raw signal as well as reduce its dimensions. Then, the Riemannian Geometry Mean (RGM) is used as the reference matrix to perform the affine transformation of the symmetric positive definite (SPD) covariance matrix calculated from the filtered signal, which makes the data from different subjects comparable. Finally, the RGC is used to obtain the result of transfer learning experiments. The proposed algorithm was evaluated on two datasets (Dataset I from real patients and Dataset II from the laboratory). By comparing with two state-of-the-art and classic algorithms in the current BCI field, Ensemble of Support Vector Machine (E-SVM) and Stepwise Linear Discriminant Analysis (SWLDA), the maximum averaged area under the receiver operating characteristic curve (AUC) score of our algorithm reached 0.836, proving the potential of our proposed algorithm.

Keywords: brain computer interface (BCI); XDAWN; Riemannian Geometry; P300; transfer learning

1. Introduction

Brain–computer interface (BCI) is a technology that allows users and computers to interact with each other through brain activity. An electroencephalogram (EEG) is used to record the brain activity under certain BCI experimental task [1]. For example, users can control the mouse on the screen to move left and right by imagining their left and right hand movements, respectively [2]. Therefore, BCI has a wide range of uses in patients with disabilities, such as patients with severe neuromuscular disease or interlocking symptoms [3,4].

Many different types of EEG signals can be used in BCI field, such as steady state visual evoked potential (SSVEP) [5], motor imagery (MI) [6], and P300 [7]. In this article, we are interested in the P300-EEG signal, which is based on event-related potentials. The P300-EEG signal is a natural response of our brain to a specific external stimulus; in response, the EEG signal will have a positive peak after

about 0.3 ms of the stimulation [8]. One of the main reasons hindering the widespread use of BCI systems is the variability of EEG signals [1,9]. Due to the variability, the feature space distribution of the EEG signal collected from different subjects or different sessions are inconsistent [10]. In addition, the BCI system requires a long calibration phase before each time it is used because, to achieve a good performance, every subject's BCI system needs to be trained by their own EEG signals and cannot use other's EEG signals [11]. One of the potential solutions to reduce or even eliminate the calibration phase is a transfer learning algorithm. In this article, we mainly study the offline transfer learning of the P300-EEG signal.

In the field of machine learning, transfer learning is defined as the ability to use the knowledge learned in a previous task or domain in a new task or domain [12]. Transfer learning in the BCI field has received extensive attention in improving the generalization performance of the classifier. Pieter-Jan et al. [13] proposed combining a Bayesian model and learning from label proportion (LLP). Gayraud et al. [14] completed a cross-session transfer of P300 data using a nonlinear transform obtained by solving an optimal transport problem and reached the highest AUC score of 0.835 for one particular subject. Lu et al. [15] proposed an adaptive classification method. The initialization of the classifier is subject-independent, and, after several minutes of online adaptation, the accuracy converges to that of a fully trained supervised subject-specific model. Morioka et al. [16] proposed to learn a dictionary of spatial filters. Some other transfer learning methods are semi-supervised learning [17] and using uniform local binary patterns [18] or artificial data generation [11]. However, one of the methods with the most potential is Riemannian Geometry method [19–21].

The Riemannian Geometry classifier is a promising and new classification method in BCI field. The main idea of Riemannian Geometry is to represent the data in the form of symmetric positive definite (SPD) covariance matrix, and then directly map the SPD covariance matrix on the Riemannian manifold. The data on the Riemannian manifold can be manipulated directly, including direct classification using the Riemannian distance. We further study the potential of this classifier in this article. Although Riemannian Geometry method has achieved many good results in the BCI field, it still has shortcomings. If the data dimension is too large, the Riemannian Geometry method will perform many calculations, which is time-consuming and causes statistical deviations [22]. Therefore, the use of Riemannian Geometry method needs to be combined with dimensionality reduction algorithms. Thus, we introduce XDAWN spatial filters. XDAWN spatial filters, specially design for Event Related Potentials (ERP), were proposed by Bertrand Rivet [23]. They can enhance the P300 component and reduce the data dimensions, which is very suitable to our needs. Then, we improve the RGC by affine transforming the SPD covariance matrix of different subjects using their own Riemannian Geometry Mean (RGM) to make the data from different subjects comparable. We finally use the Riemannian Geometry classifier to complete our transfer learning experiments on the P300-speller paradigm.

Naturally, the performance of a transfer learning algorithm largely depends on the relevance of the two tasks. For example, the P300-speller task performed between two different subjects will be more relevant than the P300 task and MI task performed on the same person. In this paper, transfer learning is defined as follows: The model is trained on Subject A and used to evaluate Subject B, where Subjects A and B come from the same dataset. The structure of this paper is as follows. Section 2 presents the introduction of the method, datasets and experiment design. Section 3 presents the experimental results. Section 4 presents discussion. Section 5 presents the conclusion.

2. Materials and Methods

2.1. Datasets

2.1.1. Dataset I

The dataset used in this study is a public dataset, that can be found on bci-horizon-2020: P300 speller with amyotrophic lateral sclerosis (ALS) patients. The experiment paradigm was proposed

by Farwell and Donchin [8]. The interface is shown in Figure 1 which is a 6×6 character matrix. They used BCI2000 [24] to collect EEG signals from eight ALS subjects. The collection process is as follows. The subject needs to type 35 characters in total. For each character, the subject looks at the character, and then each row and column flash randomly one at a time as a round (a total of 12 flashes, and 2 flashes include the target stimulus), and there are 10 rounds (in order to average the signal to reduce noise). The signal acquisition frequency is 256 Hz, bandpass filtering is from 0.1 to 30 Hz, and eight channels are used for acquisition (Fz, Cz, Pz, Oz, P3, P4, PO7, and PO8). Therefore, for each subject, we can get 420 samples (70 P300 samples and 350 non-P300 samples).



Figure 1. P300-speller interface of Dataset I.

2.1.2. Dataset II

We collected EEG signals from 10 healthy subjects, five men and five women. The stimulus interface (Figure 2) is a 4×10 matrix (including 26 English letters, 10 numbers, and 4 commonly used symbols). The collection process is as follows. All 40 characters flash randomly one time as a round (a total of 40 flashes, and 1 flash includes the target stimulus), a round lasts 1.2 s, a total of 10 rounds are performed (convenient for averaging and reducing noise), and one typing is completed. Each subject needs to type 30 characters. Sampling frequency is 250 Hz, bandpass filter is 0.1–60 Hz, and 32-channel electrode caps are used. Thus, we have data from 10 subjects, each with 1200 samples (30 P300 samples and 1170 non-P300 samples).

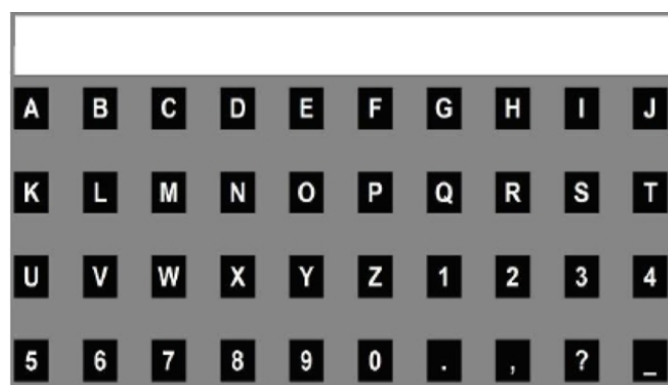


Figure 2. P300-speller interface of Dataset II.

2.2. Methods

First, we briefly introduce the XDAWN spatial filter and Riemannian Geometry classifier and propose the improved affine transformation. Then, we give the whole framework of our algorithm. Finally, we present the preprocessing details.

2.2.1. XDAWN Spatial Filter

XDAWN is a spatial filter used to find a transformation that can improve the signal-to-noise ratio and reduce the dimension of the data. The specific process is as follows: The EEG signal containing the P300 component is expressed as $X \in R^{n \times d}$, where n represents the feature dimension and d represents the number of channels of the EEG signal. We need to find the projections $W \in R^{n \times f}$, where f represents the number of filters for projection. The data filtered by this filter are $\tilde{X} = XW$. We suppose a real P300 signal $A \in R^{e \times d}$, where e represents the length of the P300 components, and we also have a noise signal $N \in R^{n \times d}$. The noise signal contains noise that conforms to a normal distribution. The position of the P300 component in the P300 signal can be given by $D \in R^{e \times n}$ through the Toeplitz matrix. Therefore, the signal can be given by $X = D^T A + N$, and the enhanced P300 signal we try to find can be expressed as $XW = DAW + NW$. We can estimate A by a least squares estimate using the pseudoinverse:

$$\hat{A} = \underset{A}{\operatorname{argmin}} \|X - DA\|_2^2 = \left(D^T D\right)^{-1} D^T X \tag{1}$$

the optimal filters W can be found by maximizing the signal-noise-ratio (SNR) as given by the generalized Rayleigh quotient [23].

$$\hat{W} = \underset{w}{\operatorname{argmax}} \frac{\operatorname{Tr}\left(W^T \hat{A}^T D^T D \hat{A} W\right)}{\operatorname{Tr}\left(W^T X^T X W\right)} \tag{2}$$

In the traditional XDAWN algorithm, we can combine QR matrix decomposition (QRD) and singular value decomposition (SVD) to solve this optimization problem. XDAWN has proven to be very effective in increasing ERPs signals [23].

2.2.2. Riemannian Geometry Classifier

The introduction of Riemannian Geometry classifier in the BCI field challenges the status of some traditional and classic classification methods. The idea of the Riemannian Geometry classifier (RGC) is to directly map data on the Riemannian manifold with measurement metrics. In this way, we can directly manipulate these data, such as averaging, stretching, and even direct classification.

The Riemannian manifold [25,26] is a non-Euclidean space, and the neighborhood of each point on the manifold is homeomorphic to Euclidean space. To simplify, a Riemannian manifold can be seen as a space that locally looks flat. The surface of the Earth can be seen as a Riemannian manifold. The reason we want to map data directly into the Riemannian manifold space is based on the assumption that, under the P300-speller task, our mental state, as well as the power and spatial distribution of the EEG signals we generate, can be considered to have a certain degree invariance, which can be encoded by the covariance matrix.

In the BCI field, when dealing with the covariance matrix we get from the EEG signal, the most commonly used matrix manifold is a symmetric positive definite (SPD) matrix manifold. Suppose we have two SPD covariance matrices $C1$ and $C2$. Both can be represented as a point on the Riemannian

manifold (Figure 3). The distance between them is called the Riemannian distance. The square of the Riemann distance can be expressed by the following equation:

$$\delta^2 (C_1, C_2) = \sum_n \log^2 \lambda_n (C_1^{-1} C_2) \tag{3}$$

where $\lambda_n(M)$ represents the n th eigenvalue of the matrix M .

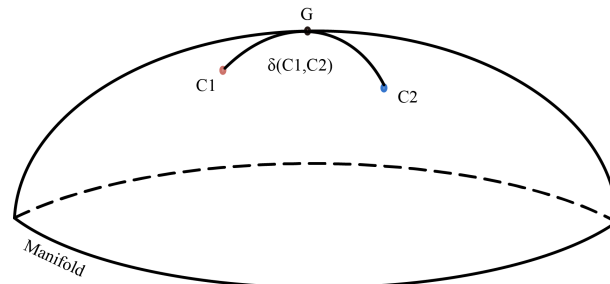


Figure 3. The distance between C_1 and C_2 is called Riemannian distance $\delta (C_1, C_2)$, where G represents the Riemannian Geometry Mean of C_1 and C_2 .

Using the Riemannian distance in Equation (3) the centroid G of a set of K SPD matrices C_1, \dots, C_K (Figure 3), also known as the Riemannian Geometry mean (RGM), can obtain the optimized solution as follows:

$$G = \operatorname{argmin}_G \sum_k \delta^2 (C_k, G) \tag{4}$$

During training, we map all the K -type SPD matrices on the Riemannian manifold and calculate the Riemannian Geometry mean of the K -type ($G_1, G_2, G_3, \dots, G_k$) (Figure 4). When new test data come in, we calculate the Riemannian distance between the point and the rest of the RGM point, and classify it as the one with the smallest Riemannian distance to the mean (RMDM).

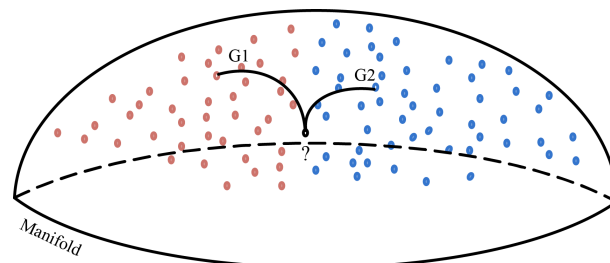


Figure 4. The Riemannian minimum distance to the mean for classification problems. Two types of Riemannian Geometry mean (G_1 and G_2) are calculated from the training data. When the data to be classified (indicated by a question mark) come in, they are classified as the ones whose Riemannian distances is smallest.

2.2.3. Affine Transformation of SPD Covariance Matrix

We propose to affine transform the SPD covariance matrix to make the data from different subjects comparable. In the Riemannian Geometry framework, the cross-subjects signal variability can be understood as the geometric transformation of the SPD covariance matrix on the Riemannian manifold. In [27], Reuderink et al. tried to solve the cross-subject variability by performing the affine transformation of the covariance matrix; however, their work did not consider the geometric structure of the covariance matrix. In [28], Baracht et al. tried to use the Riemannian Geometry framework to perform cross-session MI transfer learning; however, their method is related to the sequence of experiment task. In this article, we propose a method that combines the ideas of Riemannian Geometry classifier and affine transformation, and specially uses the Riemannian Geometry mean to complete

the selection of the reference matrix for the affine transformation. We know that the distribution of the covariance matrix of different subjects on the manifold is inconsistent, but there is a certain reference state; thus, as long as we find this reference state and express it in the form of matrix, through affine transformation, data from different subjects become comparable. We estimate a reference matrix for each subject's data, and then we use these reference matrix to perform an affine transformation on their data. After this transformation, the Riemann distance and geometric structure of the SPD covariance matrix on the manifold will not be changed. Although the reference matrix is different for every subject, it will cause the covariance matrix on the manifold to move in different directions, but, as long as a common and stable reference matrix is found, the data from different subjects will move in the same direction on the manifold and become comparable. The Riemannian Geometry Mean is a good choice to represent this reference matrix. We use Riemannian Geometry mean to calculate the reference matrix R_i ($i = 1, 2, \dots, k$.) for each subject, respectively. The affine transformation can be represented as follows:

$$C_i = R_i^{-\frac{1}{2}} C_i R_i^{-\frac{1}{2}} \tag{5}$$

C_i represents SPD covariance matrix of the i th subject.

2.2.4. Algorithm of XDAWN + RGC

The overall steps of our algorithm are as follows (Figure 5). Training stage: We use signal X_n and label Y_n as the input of the filter XDAWN, calculate the SPD covariance matrix of the filtered signal X_n^* , perform affine transformation on the SPD covariance matrix, and then use the transformed SPD covariance matrix to train the RGC classifier. Testing stage: When a new test set X_p comes in, the previously-trained XDAWN is used to perform spatial filtering to obtain X_p^* ; then, the SPD covariance matrix is obtained and affine transformed according to X_p^* and its own RGM, respectively; and, finally, it is classified using RMDM.

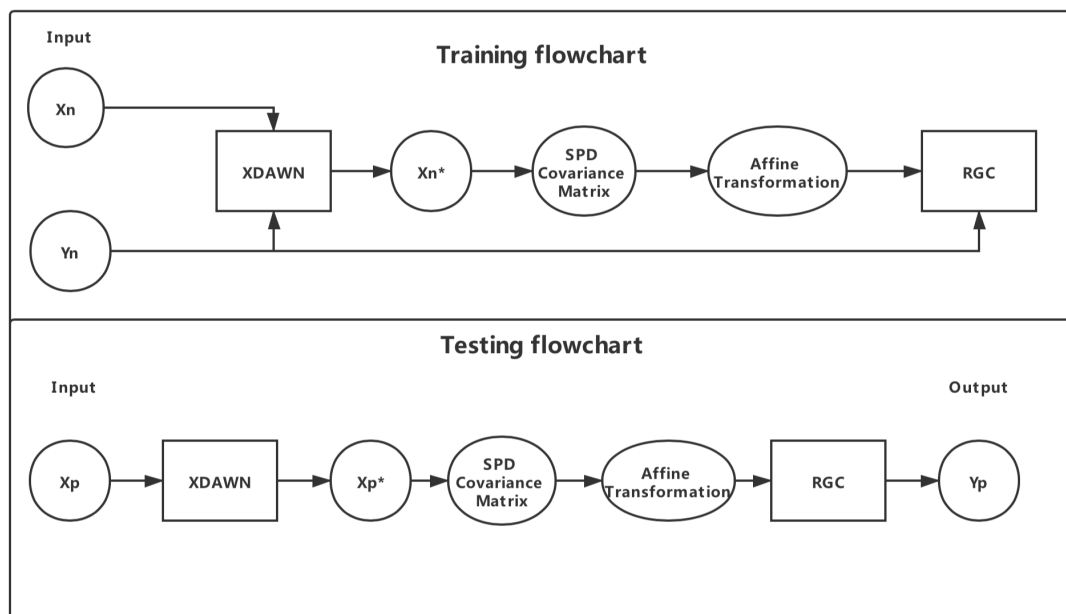


Figure 5. Flowchart of XDAWN + RGC transfer learning algorithm.

2.2.5. Data Preprocessing

In the experiment of Dataset I, rows and columns are flashed. In the experiment of Dataset II, characters are flashed. We selected the data from 0–0.5 s after flashing as a sample. We used a fifth-order Butterworth filter [29], 0.1–20 Hz bandpass filtering, and performed down sampling to 34 Hz. The data structure for each subject of Datasets I and II can be expressed as follows:

For Dataset I: $X_i^{420 \times 8 \times 17}$ ($i = 1, 2, \dots, 8$) (420 represents the number of samples, 8 represents the number of channels, and 17 represents feature dimensions).

For Dataset II: $X_i^{1200 \times 32 \times 17}$ ($i = 1, 2, \dots, 10$) (1200 represents the number of samples, 32 represents the number of channels, and 17 represents feature dimensions).

2.3. Experiment

2.3.1. Experiment 1: One-to-One Transfer Learning

In the first experiment (Table 1), we performed one-to-one transfer learning. We selected one subject as the test set, and the other subjects took turns as training set. The results were averaged to get the final result.

Table 1. Algorithm of experiment 1.

<ol style="list-style-type: none"> 1. Leave one subject's data X_i ($i = 1, 2, \dots, k$) for testing purpose. 2. For s in k: 3. Input training data X_S, label Y_S ($s = 1, 2, \dots, k$, except i). 4. Calculate X_S^* after XDAWN filtering. 5. Calculate SPD covariance matrix M_S according to X_S^*. 6. Calculate the Riemannian Geometry mean point G, as reference matrix and use G to affine transform the M_S to get M_S^*. 7. Map M_S^* to the Riemannian manifold, and calculate the Riemannian Geometry mean points for two classes, $G1$ and $G2$. 8. End for 9. Input test data X_i. 10. After XDAWN filtering, calculate its SPD covariance matrix and affine transform it and use RGC classifiers to classify. 11. Output the class with the smallest Riemann distance.
--

2.3.2. Experiment 2: All-to-One Transfer Learning

In the second experiment (Table 2), we used the leave-one-out method to evaluate the performance of the classifier in the case of using a large training set. We selected one subject as testing data, and the remaining subjects were used as training data. More data mean more complete feature space, thus we used Bootstrap Aggregating (BA) method for Experiment 2. BA method was proposed by Breiman et al. [30] and is often used in BCI [31,32]. We trained k classifiers for k subjects (one subject for one classifier), and the final result was voted by all classifiers, that is, the result receiving the largest number of votes.

Table 2. Algorithm of experiment 2.

<ol style="list-style-type: none"> 1. Leave one subject's data X_i ($i = 1, 2, \dots, k$) for testing purpose. 2. for s in $k-1$: 3. Input training data set $X = X_s$ ($s = 1, 2, \dots, k$, except i), label Y 4. Calculate X^* after XDAWN filtering 5. Calculate the SPD covariance matrix M of X^*. 6. Calculate the Riemannian Geometry mean point G, as reference matrix and use G to affine transform the M to get M^*. 7. Project M^* into the Riemannian manifold, and calculate the Riemannian Geometry mean points for two classes, $G1$ and $G2$. 8. end for 9. Input test data X_i and After XDAWN filtering, calculate its SPD covariance matrix and affine transform it and use RGC classifiers to classify. 10. Output the result by the largest votes.

3. Results

Due to the unbalance of the samples classes, we used the area under the receiver operating characteristic curve (AUC) [33] as our performance metric. Stepwise Linear Discriminant Analysis (SWLDA) [34] and Ensemble of Support Vector Machines (E-SVM) [35] are state-of-the-art classifiers in the current statistical classifiers used in BCI. We compared our proposed XDawn + RGC method with these two methods to obtain a fair result. SWLDA performs stepwise model selection before applying conventional linear discriminant analysis, reducing the number of features used for classification. E-SVM uses multiple SVM classifiers to make decisions together. The results of the two experiments are presented below.

3.1. One-to-One Transfer Learning

It can be seen in Tables 3 and 4 that, in the one-to-one transfer learning experiment, the improvement of our proposed method XD + RGC compared to E-SVM and SWLDA is obvious, and compared with RGC alone our method still has slight improvement. The average AUC score in Dataset I reached 0.776 and the largest value is 0.821. The average AUC score in Dataset II reaches 0.787 and the maximum value is 0.813. As can be seen in Figure 6, our method has a good overall stability and does not have much fluctuation.

Table 3. One-to-One transfer learning results of Dataset I. The first row represents the testing subject. The second and third rows represent the AUC score results obtained using the classic E-SVM and SWLDA methods. The fourth row represents the results obtained by our proposed XDawn + RGC method. The last column represents the average AUC value of all subjects.

Testing Subjects	S1	S2	S3	S4	S5	S6	S7	S8	Avg.
E-SVM	0.512	0.532	0.589	0.503	0.537	0.612	0.535	0.563	0.547
SWLDA	0.534	0.525	0.523	0.578	0.549	0.562	0.603	0.512	0.548
RGC	0.703	0.766	0.721	0.734	0.743	0.792	0.805	0.702	0.746
XD + RGC	0.744	0.793	0.758	0.772	0.789	0.815	0.821	0.716	0.776

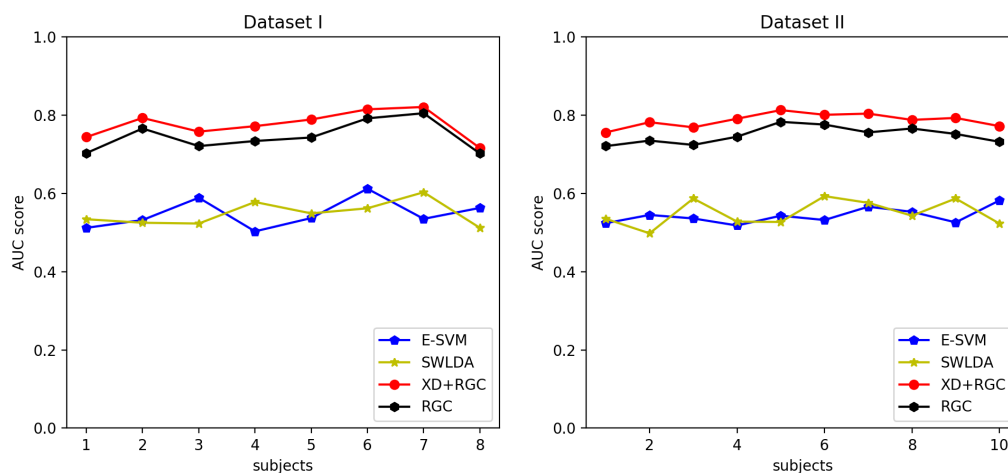


Figure 6. One-to-One transfer learning result of: Dataset I (left); and Dataset II (right).

Table 4. One-to-One transfer learning AUC score results of Dataset II.

Testing Subjects	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Avg.
E-SVM	0.524	0.545	0.536	0.518	0.543	0.532	0.566	0.553	0.526	0.582	0.542
SWLDA	0.535	0.521	0.587	0.528	0.527	0.593	0.576	0.543	0.587	0.523	0.552
RGC	0.721	0.735	0.724	0.745	0.783	0.776	0.756	0.766	0.752	0.732	0.749
XD+RGC	0.756	0.782	0.769	0.791	0.813	0.801	0.804	0.788	0.793	0.772	0.787

3.2. All-to-One Transfer Learning

It can be seen in Tables 5 and 6 that, in the all-to-one transfer learning, the average AUC value of our method in Dataset I reached 0.836 and the maximum value is 0.879, while the average AUC value in Dataset II reached 0.830 and the maximum value is 0.865. It can be seen in Figure 7 that our method is still stable overall, and, compared to one-to-one transfer learning, we use the BA method obtain higher results, because the BA method can effectively use a more complete feature space when the number of data increases. The result of RGC is still slightly lower than XDAWN + RGC, which proves that XDAWN can increase the performance of RGC.

Table 5. All-to-One transfer learning AUC score results of Dataset I. The first and second rows represent the results obtained using the classic E-SVM and SWLDA methods. The third row (XD + RGC) represents the results obtained by our proposed XDAWN + RGC method. The last column represents the average AUC value of all subjects.

Testing Subjects	S1	S2	S3	S4	S5	S6	S7	S8	Avg.
E-SVM	0.643	0.628	0.633	0.672	0.557	0.663	0.642	0.647	0.636
SWLDA	0.642	0.657	0.632	0.599	0.643	0.682	0.752	0.634	0.655
RGC	0.742	0.789	0.772	0.801	0.822	0.831	0.812	0.833	0.800
XD+RGC	0.789	0.823	0.794	0.843	0.864	0.852	0.842	0.879	0.836

Table 6. All-to-One transfer learning AUC score results of Dataset II.

Testing Subjects	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Avg.
E-SVM	0.638	0.654	0.623	0.641	0.552	0.732	0.662	0.683	0.687	0.632	0.650
SWLDA	0.653	0.642	0.662	0.582	0.689	0.701	0.743	0.632	0.621	0.674	0.660
RGC	0.763	0.801	0.763	0.821	0.842	0.821	0.827	0.801	0.811	0.808	0.806
XD+RGC	0.793	0.836	0.788	0.842	0.865	0.842	0.862	0.827	0.824	0.825	0.830

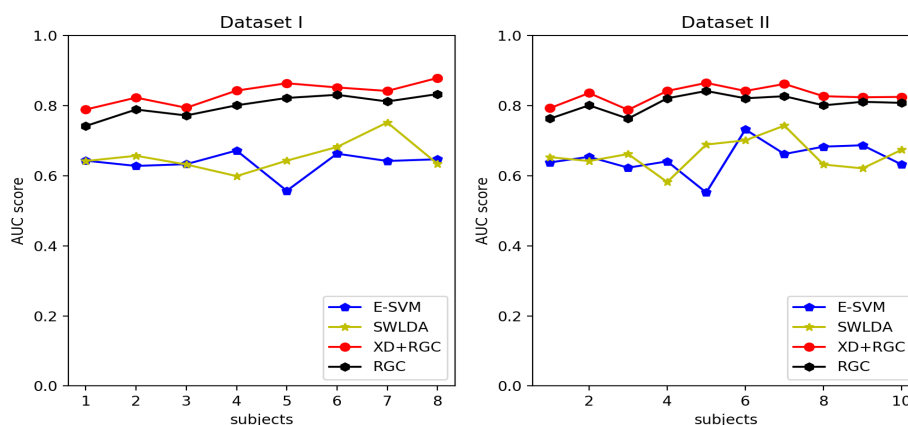


Figure 7. All-to-One transfer learning result of: Dataset I (left); and Dataset II (right).

4. Discussion

This paper proposes a XDAWN + RGC transfer learning algorithm for P300-EEG signal. The XDAWN spatial filter can effectively improve the quality of the evoked P300 components by

considering the signal and noise simultaneously. XDAWN also greatly reduced the feature dimensions for the subsequent Riemannian Geometry classifier and improves the performance of the Riemannian Geometry classifier. After mapping the covariance matrix to the Riemannian manifold space, we first performed an affine transformation on the covariance matrix, so that data from different subjects move in the same direction on the Riemannian manifold, making the data comparable without changing the Riemannian distance and geometry structure of the data. There are several reasons for promoting the use of Riemannian Geometry classifier. Due to its logarithmic nature, the Riemann distance is robust to extreme values (noise). Moreover, the Riemannian distance of the SPD matrix is invariant to the matrix inversion and any linearly invertible transformation of the matrix [36]. These characteristics partially explain why the Riemann classification provides good generalization capabilities.

From the results of two experiments, it is proved that our proposed method has greatly improved the transfer learning algorithm’s performance compared with two classic classification methods, E-SVM and SWLDA. The highest average AUC value reached 0.836, and it also proved that, with the small number of available data in Experiment 1, our proposed transfer learning method can already achieve a fairly good performance. We visualized the data of two subjects from the two datasets, respectively, for a more intuitive understanding of the affine transformation.

From the visualization of the two datasets (Figures 8 and 9), we can see that the covariance matrix after the affine transformation is more concentrated and consistent in spatial distribution, which proves that our proposed affine transformation is effective.

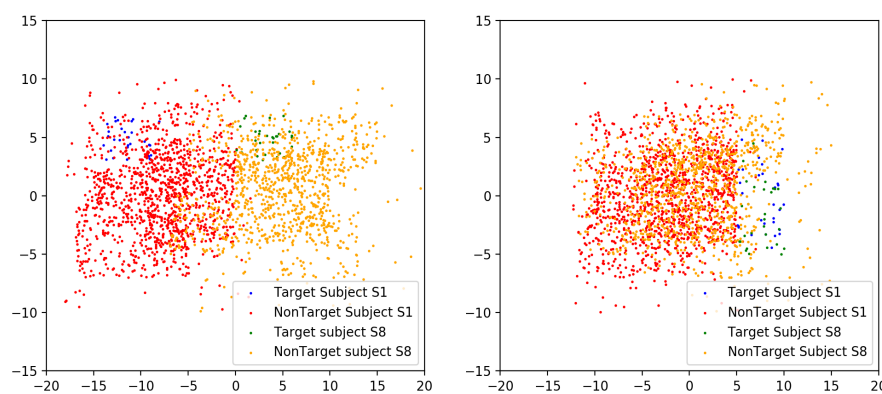


Figure 8. SPD covariance matrix of Subjects S1 and S8 before affine transformation of Dataset I (left); and SPD covariance matrix of Subjects S1 and S8 after affine transformation of Dataset I (right).

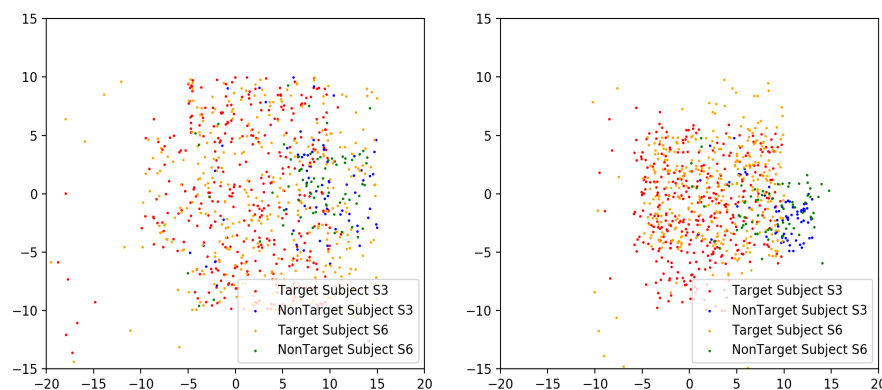


Figure 9. SPD covariance matrix of Subjects S3 and S6 before affine transformation of Dataset II (left); and SPD covariance matrix of Subjects S3 and S6 after affine transformation of Dataset II (right).

The reason the overall performance is good and stable is that using the covariance matrix to represent the data can better capture the correlation between features; mapping these covariance matrices on the Riemannian manifold as points, the geometric structure of these points will be demonstrated clearly. Affine transformation can be performed on the data without changing the geometric properties of the data, and the Riemannian Geometry mean is used to represent the reference matrix. It is considered that, under the P300 task, the subject's mental state is in a relatively stable state. We use Riemannian Geometry mean of all the samples to capture this stable state. In addition, the Riemannian Geometry classifier has no parameters to train. We use XDAWN to enhance the P300 signal while reducing the data dimension, which greatly reduces the computational expenditure of the Riemannian Geometry classifier.

5. Conclusions

In this work, we show that an algorithm combining XDAWN and Riemannian Geometry classifiers can be used for cross-subject transfer learning to improve the generalization ability on P300-EEG signals. In particular, we propose to affine transform the SPD covariance matrix from different subjects using their own Riemannian Geometry mean as the reference matrix, respectively, before classifying. From our results, our method has the potential to reduce the calibration phase or even eliminate it, especially when the amount of data that can be used as training data are few. Overall, it is time to change the golden standard classification method used in EEG-based BCI. We could change the focus from a classic SWLDA or SVM design to a Riemannian geometric classifier. Our future work will focus on the development of a more robust BCI transfer learning algorithm that still has good performance and can be used online. We aim to mix our algorithm with some other algorithms [37–39] in future work. We believe that the transfer learning algorithm related to Riemannian Geometry has a promising future.

Author Contributions: Investigation, X.L.; Methodology, Y.X.; Supervision, F.L.; Validation, F.H.; Writing (original draft), Y.X.; Writing (review and editing), F.W. and D.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (Grant No. 61906019), the Natural Science Foundation of Hunan Province, China (Grant No. 2019JJ50649), the Scientific Research Fund of Hunan Provincial Education Department (Grant Nos. 18C0238 and 19B004), the "Double First-class" International Cooperation and Development Scientific Research Project of Changsha University of Science and Technology (No. 2018IC25), and the Young Teacher Growth Plan Project of Changsha University of Science and Technology (No. 2019QJCZ076).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Clerc, M. Brain Computer Interfaces, Principles and Practise. *Biomed. Eng. Online* **2013**, *12*, 1–4.
2. Wolpaw, J.R.; McFarland, D.J.; Neat, G.W.; Forneris, C.A. An EEG-based brain-computer interface for cursor control. *Electroencephalogr. Clin. Neurophysiol.* **1991**, *78*, 1–259. [[CrossRef](#)]
3. Birbaumer, N. Brain-computer-interface research: Coming of age. *Clin. Neurophysiol.* **2006**, *117*, 479–482. [[CrossRef](#)] [[PubMed](#)]
4. Hochberg, L.R.; Serruya, M.D.; Friehs, G.M.; Mukand, J.A.; Saleh, M.; Caplan, A.H.; Branner, A.; Chen, D.; Penn, R.D.; Donoghue, J.P. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature* **2006**, *442*, 164–171. [[CrossRef](#)]
5. Lin, Z.; Zhang, C.; Wu, W.; Gao, X. Frequency Recognition Based on Canonical Correlation Analysis for SSVEP-Based BCIs. *IEEE Trans. Biomed. Eng.* **2006**, *53*, 2610–2614. [[CrossRef](#)]
6. Pfurtscheller, G.; Neuper, C. Motor imagery and direct brain-computer communication. *Proc. IEEE* **2001**, *89*, 1123–1134. [[CrossRef](#)]
7. Polich, J. Updating P300: An integrative theory of P3a and P3b. *Clin. Neurophysiol.* **2007**, *118*, 2128–2148. [[CrossRef](#)]

8. Farwell, L.A.; Donchin, E. Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalogr. Clin. Neurophysiol.* **1988**, *70*, 1–523. [[CrossRef](#)]
9. Blankertz, B.; Sannelli, C.; Halder, S.; Hammer, E.M.; Kübler, A.; Müller, K.R.; Curio, G.; Dickhaus, T. Neurophysiological predictor of SMR-based BCI performance. *Neuroimage* **2010**, *51*, 1303–1309. [[CrossRef](#)]
10. Clerc, M.; Daucé, E.; Mattout, J. *Adaptive Methods in Machine Learning*; John Wiley & Sons: Hoboken, NJ, USA, 2016.
11. Lotte, F. Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain–Computer Interfaces. *Proc. IEEE* **2015**, *103*, 871–890. [[CrossRef](#)]
12. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [[CrossRef](#)]
13. Pieter-Jan, K.; Martijn, S.; Benjamin, S.; Klaus-Robert, M.; Michael, T.; Marco, C. True Zero-Training Brain–Computer Interfacing—An Online Study. *PLoS ONE* **2014**, *9*, e102504.
14. Gayraud, N.T.; Rakotomamonjy, A.; Clerc, M. Optimal Transport Applied to Transfer Learning for P300 Detection; In *7th Graz Brain–Computer Interface Conference*; Springer: Graz, Austria, 2017.
15. Lu, S.; Guan, C.; Zhang, H. Unsupervised Brain Computer Interface Based on Intersubject Information and Online Adaptation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2009**, *17*, 135–145. [[PubMed](#)]
16. Morioka, H.; Kanemura, A.; Hirayama, J.I.; Shikauchi, M.; Ogawa, T.; Ikeda, S.; Kawanabe, M.; Ishii, S. Learning a common dictionary for subject-transfer decoding with resting calibration. *NeuroImage* **2015**, *111*, 167–178. [[CrossRef](#)] [[PubMed](#)]
17. Li, Y.; Guan, C. A Semi-supervised SVM Learning Algorithm for Joint Feature Extraction and Classification in Brain Computer Interfaces. *IEEE Eng. Med. Biol. Soc.* **2006**, *1*, 2570–2573.
18. Zhang, D.; Gaobo, Y.; Feng, L.; Jin, W.; Kumar, S.A. Detecting seam carved images using uniform local binary patterns. In *Multimedia Tools & Applications*; Springer: Berlin, Germany, 2018.
19. Zanini, P.; Congedo, M.; Jutten, C.; Said, S.; Berthoumieu, Y. Transfer Learning: A Riemannian Geometry framework with applications to Brain–Computer Interfaces. *IEEE Trans. Biomed. Eng.* **2017**, *65*, 1107–1116. [[CrossRef](#)]
20. Congedo, M.; Barachant, A.; Bhatia, R. Riemannian Geometry for EEG-based brain-computer interfaces; A primer and a review. In *Brain–Computer Interfaces*; Taylor & Francis: Abingdon, UK, 2017; pp. 1–20.
21. Yger, F.; Berar, M.; Lotte, F. Riemannian Approaches in Brain–Computer Interfaces: A Review. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 1753–1762. [[CrossRef](#)]
22. Lotte, F.; Bougrain, L.; Cichocki, A.; Clerc, M.; Congedo, M.; Rakotomamonjy, A.; Yger, F. A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update. *J. Neural Eng.* **2018**, *15*, 031005. [[CrossRef](#)]
23. Rivet, B.; Souloumiac, A.; Attina, V.; Gibert, G. xDAWN Algorithm to Enhance Evoked Potentials: Application to Brain–Computer Interface. *IEEE Trans. Biomed. Eng.* **2009**, *56*, 2035–2043. [[CrossRef](#)]
24. Schalk, G.; McFarland, D.J.; Hinterberger, T.; Birbaumer, N.; Wolpaw, J.R. BCI2000: A general-purpose brain-computer interface (BCI) system. *IEEE Trans. Biomed. Eng.* **2004**, *51*, 1034–1043. [[CrossRef](#)]
25. Lang, S. *Differential and Riemannian Manifolds*; Springer: Berlin, Germany, 2012; p. 160.
26. Absil, P.A.; Mahony, R.; Sepulchre, R. *Optimization Algorithms on Matrix Manifolds*; Princeton University Press: Princeton, NJ, USA, 2008.
27. Reuderink, B.; Farquhar, J.; Poel, M.; Nijholt, A. A subject-independent brain-computer interface based on smoothed, second-order baselining. In Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society, Boston, MA, USA, 30 August–3 September 2011.
28. Barachant, A.; Bonnet, S.; Congedo, M.; Jutten, C. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing* **2013**, *112*, 172–178. [[CrossRef](#)]
29. Selesnick, I.W.; Burrus, C.S. Generalized digital Butterworth filter design. *IEEE Trans. Signal Process.* **1998**, *46*, 1688–1694. [[CrossRef](#)]
30. Breiman, L. Bagging predictors. In *Machine Learning*; Springer: Berlin, Germany, 1996; Volume 24, pp. 123–140.
31. Blankertz, B.; Dornhege, G.; Müller, K.R.; Schalk, G.; Krusienski, D.; Wolpaw, J.R.; Schlogl, A.; Graimann, B.; Pfurtscheller, G.; Chiappa, S.; et al. Results of the BCI Competition III. In *BCI Meeting*; Elsevier: Amsterdam, The Netherlands, 2005.
32. Sun, S.; Zhang, C.; Zhang, D. An experimental evaluation of ensemble methods for EEG signal classification. In *Pattern Recognition Letters*; Elsevier: Amsterdam, The Netherlands, 2007; Volume 28, pp. 2157–2163.

33. Lobo, J.M. AUC: A misleading measure of the performance of predictive distribution models. *Glob. Ecol. Biogeogr.* **2007**, *17*, 145–151. [[CrossRef](#)]
34. Krusienski, D.J.; Sellers, E.W.; Cabestaing, F.; Bayoudh, S.; McFarland, D.J.; Vaughan, T.M.; Wolpaw, J.R. A comparison of classification techniques for the P300 Speller. *J. Neural Eng.* **2006**, *3*, 299–305. [[CrossRef](#)]
35. Rakotomamonjy, A.; Guigue, V. BCI competition III: Dataset II—Ensemble of SVMs for BCI P300 speller. *IEEE Trans. Biomed. Eng.* **2008**, *55*, 1147–1154. [[CrossRef](#)]
36. Lin, T.; Zha, H. Riemannian Manifold Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 796.
37. Xiang, L.; Guo, G.; Yu, J. A convolutional neural network-based linguistic steganalysis for synonym substitution steganography. *Math. Biosci. Eng.* **2020**, *18*, 1041–1058. [[CrossRef](#)]
38. Liu, Z.; Lai, Z.; Ou, W.; Zhang, K.; Zheng, R. Structured optimal graph based sparse feature extraction for semi-supervised learning. *Signal Process.* **2020**, *170*, 107456. [[CrossRef](#)]
39. Zeng, D.; Dai, Y.; Li, F.; Wang, J.; Sangaiah, A.K. Aspect based sentiment analysis by a linguistically regularized CNN with gated mechanism. *J. Intell. Fuzzy Syst.* **2019**, *36*, 3971–3980. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).