Brain-Voyant: A General Purpose Machine-Learning Tool for Real-Time fMRI Whole-Brain Pattern Classification

Ori Cohen*‡, Rafael Malach†, Moshe Koppel* and Doron Friedman‡
*Department of Computer Science, Bar-Ilan University Ramat-Gan, 52900 Israel.
†Department of Neurobiology, Weizmann Institute of Science, Rehovot 76100, Israel.
‡Advanced Reality Lab, The Interdisciplinary Center Herzliya (IDC H.), P.O. Box 167 Herzliya, 46150, Israel.
Corresponding authors: orioric@gmail.com, doronf@idc.ac.il

***Abstract*—We have developed Brain-Voyant, an efficient general-purpose machine learning tool for real-time functional magnetic resonance imaging classification using whole-brain data, which can be used to explore novel brain-computer interface paradigms or advanced neurofeedback protocols. We have created a convenient and configurable front-end tool that receives fMRI-based multi-voxel raw brain data as input. Our tool processes, analyses, classifies and transfers the classification to an external object such as a virtual avatar or a humanoid robot in real-time. Our tool is focused on minimizing delay time, and to that end, it employs a method that is based on examining in advance the voxels that have been found to be task-relevant in the machine learning model training phase.**

The tool's code base was designed to be easily extended to support additional feature reduction, normalization and classification algorithms. This tool was used in several published studies using motor execution, motor imagery, and visual category classification in cue-based and free-choice brain-computer interface experiments, with both healthy and amputated subjects. This tool is not limited by number of classes, is not limited to predefined regions of interest, and classifier instances can run in parallel to combine multiple classification tasks in real time. Finally, our tool is able use the slow peaking blood-oxygen-level dependent signal to classify our subjects' intention during the two-second window TR. We release this tool as open-source for non-commercial usage.

I. INTRODUCTION

Brain-Voyant is based on statistical machine learning classification of subjects' brain state in real time and converges on task-relevant voxels based on whole-brain activity. We automatically classify subjects' intentions in a single session and in real time, and transmit their intentions to an external device (avatar or a humanoid robot), which allows the subject in the functional magnetic resonance imaging (fMRI) scanner to perform tasks in a virtual environment or in the physical world. We built Brain-Voyant to be efficient and robust in terms of usability and processing time, to be flexible in using multiple machine learning schemes, and to be used in offline or real-time fMRI brain-computer interface (BCI).

Real time fMRI is a promising risk-free non-invasive method for next generation neurofeedback (NF), rehabilitation, rapid functional mapping, and basic neuroscience. Due to its superior spatial resolution, fMRI may be used to classify a much wider set of mental patterns than electroencephalogram (EEG) and may thus facilitate exploring new BCI paradigms. FMRI-based BCI paradigms can be made to localize underlying brain patterns and transfer the paradigms to other, more accessible, signals, such as EEG (see [1]) and near-infrared spectography (NIRS) [2]. There is also compelling evidence that fMRI-based NF can have desired behavioral outcomes [3]. It can be used for training paralyzed patients before undergoing surgery for invasive BCI. In all these cases, the high spatial resolution and whole-brain coverage allows us to target and analyse very specific brain areas, which are unknown beforehand – making fMRI BCI vital for clinical populations. For example, our system allowed amputees control of an external device with high accuracy using their missing hand [4].

It has been argued that the fMRI raw-signal cannot be used for real-time BCI due to the sluggish nature of the blood-oxygen-level dependent (BOLD) signal, in the range of 6 seconds delay [5]. However, we have demonstrated that participants are able to rapidly adapt to the long delays, and by likely developing a predictive strategy, achieve a remarkable level of successful performance [6].

The last few years have established the superiority of multivariate pattern analysis (MVPA) and machine learning (ML) over univariate analysis of discrete regions of interest (ROI) of fMRI data [7], [8], and it is clear that many perceptual, cognitive, or emotional states generally recruit a distributed network of brain regions rather than single locations. However, almost all real-time fMRI studies so far are based on feedback from signals derived from single ROIs [9], [10]. Similar studies have already shown the possibility and advantages of data-driven, ML-based ROI selection for rt-fMRI BCI [11], [12], but an ROI-based method will not work as well when there is no prior knowledge about the brain patterns expected, e.g., following brain injury or amputation [4], i.e., when the participating voxels expand or move to other non-related brain regions. These patterns are also individual as seen in Figure 1. Other tools are less flexible in terms of normalization, feature selection and classification algorithms [13].

Despite years of research on motor imagery EEG-based BCI, there is only one study demonstrating 4 class, self-paced navigation [14]. While it is impossible to compare performance for a free choice task, the cue-based accuracy indicates that our fMRI-BCI accuracy is much higher than those achieved with EEG; in Scherer et al.'s study the top 3 subjects (selected from 8 candidates), reached levels of 71-86% accuracy after three training sessions. Our results indicate that all subjects can reach high accuracy (90%-95%) after a

978-1-5090-6014-6/18/$31.00 ©2018 IEEE

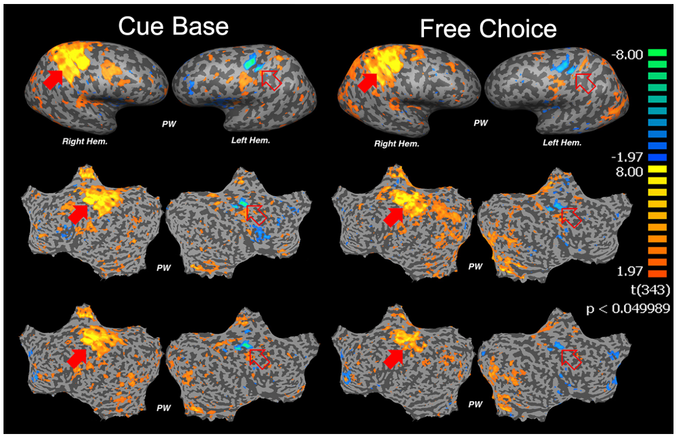

Fig. 1. A gallery visualization of the left vs right contrast using a p<0.05 for the cue-based and free choice tasks, for the amputee group

few trials.

Recently, there have been some first attempts at using real-time fMRI for BCI, based on different mental tasks, such as motor imagery, mental calculation, and inner speech [15], using covert visual attention [16] and using motor imagery to control a robotic arm [17]. We have also implemented a real time BCI fMRI tool based on manually selected ROIs for controlling an avatar [6] and a humanoid robot [18], in which a baseline for ROI-based BCI performance and speed was established.

There are several studies including online fMRI ML-based methods, using online support vector machine (SVM) classification [19], controlling a robotic arm with 24.3 seconds per decision [20], using predefined ROIs [21], [22], and binary classification of left and right hand fingers with only 80% classification accuracy. Our tool is designed to inspect the entire brain and find relevant voxels without relying on predefined ROIs, which allows for faster intention-detection, can go beyond binary classification scheme, and can also detect covert states such as "rest".

In previous papers [4], [6], [18] we have reported experimental results of high accuracy BCI control in various schemes, demonstrating some of the advantages of our approach. In this paper we provide an overview of the technical system and method behind these studies, explaining how our approach can be used for a wide range of for real-time fMRI ML-based research. Additionally, this tool is made available to the BCI community[1].

II. THE SYSTEM

A. fMRI scanner

Imaging was performed on a 3T Trio Magnetom Siemens scanner, and all images were acquired using a 12 channel head matrix coil. Three-dimensional T1-weighted anatomical scans were acquired with high resolution 1-mm slice thickness, 3D MP-RAGE sequence, repetition time (TR) 2300ms, TE 2.98ms, $1mm^3$ voxels). For BOLD scanning, T2*-weighted images using echo planar imaging sequence (EPI) were acquired using the following parameters: TR 2000ms, TE 30ms, Flip angle 80, 35 oblique slices without gap, 20 towards coronal plane from Anterior Commissure-posterior Commissure (ACPC), $3\times3\times4$mm voxel size, covering the whole cerebrum. Visual feedback is provided by a mirror, placed 11cm from the eyes of the subject and 97.5cm from a screen, leaving 108.5cm from the screen to the eyes of the subject. There is a trade-off between the scanning rate and the number of slices scanned, and we have opted for a scan time (TR) of 2000ms. Using our scanner, a lower TR of 1000ms is possible if we scan a smaller portion of the brain (i.e., less slices), but this would prevent us from performing a full brain analysis in the future.

B. Implementation

The tool was coded in Microsoft Visual Studio C#, running on Microsoft Windows 7. The minimum requirements are a quad-core Intel i7 and 32 GB RAM. For feature selection and classification we use Weka's machine learning algorithms API [23] that can be easily used to extend our tool. Our tool was also integrated with external devices such as humanoid robots and the Unity 3D game engine[2], allowing a virtual environment feedback for engaging subjects in a wide range of scenarios and tasks.

Training and applying classifiers in real-time requires that learning be executed faster than is generally done in the application of ML to fMRI. Our tool is optimized for memory usage, processing speed, and classification speed. To achieve faster processing, we focus on several areas: feature filtering, feature selection and removal of redundant samples. In addition, we have used implementation techniques including minimization of computational cycles and RAM consumption, by using sparse data structures, using RAM instead of disk access, and transferring data between processes by using an inter-process communication method.

C. fMRI data preprocessing

Dicom files[3] from the scanner are received and preprocessed by Turbo Brain Voyager (TBV)[4]. : spatial Gaussian smoothing is applied, and they are auto aligned by a real-time algorithm that uses a statistical atlas to automatically position the scanned slices [24] and applied with a real-time three-dimensional motion correction algorithm, the prospective acquisition correction (PACE) algorithm, which adjusts slice position and orientation in order to reduce motion artifacts [25]. This pre-processing is applied at the initial scan, between every two scans for subsequent subject movement, and when subjects return for additional scans, on different days.

Additionally, we created a plugin for TBV that transmits the fMRI recordings to our tool in real-time, using an inter-process communication method. Each recorded data sample

[1] Brain-Voyant,https://github.com/orico/Brain-Voyant

[2] Unity Technologies, California, http://unity3d.com/

[3] http://medical.nema.org/

[4] BrainInnovation,Netherlands,http://www.brainvoyager.com/

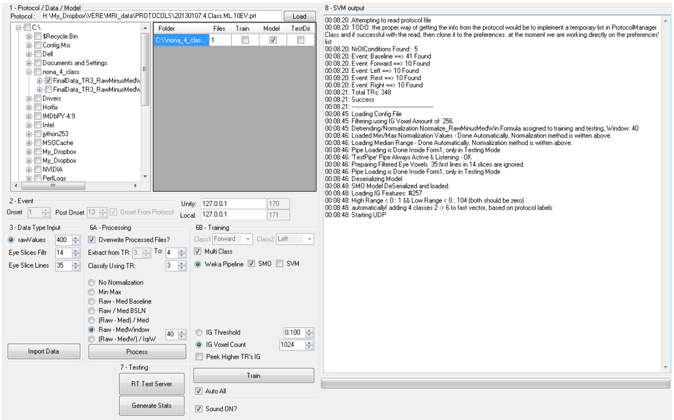

Fig. 2. The graphical user interface that allows ML training and testing under the Windows platform.

is a 3D matrix of the entire brain area that is composed of 204,800 voxels, which hold the raw BOLD-derived values. The 3D matrix is flattened into a 1-dimensional vector and transmitted to our tool.

D. The Data

We record several runs for each subject, each recorded run is composed of a 30-45 trials, and for each trial the data is composed of 8 or 9 samples that correspond to the brain scans in TRs 1 to 8-9 following the trial. We assign every data sample with a label according to the balanced and predefined experimental protocol. As a preliminary stage we have selected one subject in order to find the optimal settings.

1) Front-end: Running a Study: The application (Figure 2) includes a complete tool for running a wide range of real-time fMRI studies with different experimental protocols, different analysis methods, and it can interface with any external device. The tool allows easily configuring classification and interaction parameters during an experiment and playing back experimental sessions. The initial parameters are embedded in the graphical user interface and were selected empirically.

The application allows the researcher to select several initial parameters such as a TBV-compatible protocol-file, a local and remote IP addresses for the external device, a threshold for the purpose of removing non brain voxels, the number of horizontal lines and scan slices that will be deleted for the purpose of removing the subject's eyes, a TR that will be used for training (i.e., where the signal peaks) and a normalization formula and window size. In the training step, the researcher selects whether to train using multi-class SVM or to manually select two classes for a binary SVM classifier. The last step is to select information gain (IG) parameters such as the top K% (threshold) or amount of voxels. During the entire process, the tool outputs information regarding the protocol: the selected parameters, progress information for the data reading, processing, feature selection and training. The configuration is then saved to an XML file, the selected feature selected voxels are saved to an XML file, the model is saved as a Weka binary model file and the output log is saved to a text file. Finally, we display the selected voxels overlaid on a map of the subject's brainand the 3D brain-image is saved automatically.

In the testing phase, the researcher loads the configuration file, which points to the protocol, parameters, feature indexes and SVM model. The application automatically enters into a data receiving mode and waits for data to be sent from TBV's plugin. When the data is received, the algorithm classifies it and sends the classification over UDP to the external object. At the end of the testing phase, the output log is saved to a text file.

E. Data Processing

1) Noise and eye filtering: Our tool can reduce the feature space by setting an activity threshold, which removes voxels that belong to the empty space around the head. Furthermore, we specify K voxel rows from the first J slices that belong to the subject's eyes, reducing the number of voxels to approximately 26-30,000 voxels per TR. Both filters remove voxels that are not part of the brain. An activity threshold of 400 and 35 vertical rows from the first 14 slices are generally chosen for all subjects.

2) Normalization: Since fMRI data tends to have non-linear non-homogeneous drifts, we introduce a normalization process that has been verified to perform well in real-time; given a raw value at voxel i and time t, $r_{i,t}$, and a sliding window of length w, we derive a new value for each raw value:

$$r'_{i,t} = r_{i,t} - median(r_{i,t-w+1}, ..., r_{i,t}) \tag{1}$$

We have empirically established that a w of 40 TRs (80 seconds) is optimal for our datasets.

3) Feature selection: We use the IG algorithm, to automatically converge on the most relevant voxels in the brain: those with highest IG to the extent that the uncertainty (entropy) regarding the class is diminished when the value of the voxel is known [26].

Since the BOLD signal peaks only 8-12 seconds (TRs 4-6) after the cue, we create TR-based datasets, containing all the data samples from a chosen TR and apply the IG feature selection to the them.

We introduced 'IG peeking': this method harnesses information that is only available at a later TR and uses it in an earlier TR in order to improve the classifier's accuracy, i.e., using influential voxels from a later TR with current TR raw-data is more informative than using current TR voxels. In practice, we use voxels that are identified by IG as influential, but we use them 2 seconds earlier, i.e., we select all the data samples from a certain TR after trials and use IG for feature selection but apply these voxels to the data samples from an earlier TR.

We systematically explored three free parameters that are available in our tool: i) choosing the TR to train on, ii) looking at a higher TR's IG indices for training at the current TR, i.e. IG 'peeking', and iii) choosing a number of top ranked IG voxels. We found that training on a later TR, such as TR-4, provided the highest accuracy. However, there is a trade-off:

eventually we want the feedback to be provided as fast as possible, and choosing a later TR might only allow accurate classification if the time between feedbacks is long. Therefore, we decided to create two datasets for the training process; the first dataset is composed of all the data samples in TR3 of each condition, and the second dataset from TR4. We used TR-4 for feature selection, selecting the top 1024 voxels, these voxels were applied on the second dataset (TR-3) for training and for real-time classification.

4) Feature engineering: We have experimented with several methods of feature engineering, which were tested off-line using chronological run-based and compared with the 'IG peeking' method: i) for each condition we incrementally concatenated data from several consecutive TRs (i.e., [TR1 & TR2], [TR1 & TR2 & TR3], or from [TR1 & TR2 & TR3 & TR4]), ii) we concatenated the difference between raw-values of adjacent TRs (i.e., [TR3 & δ(TR3-TR2) & δ(TR2-TR1)]), and iii) by spatial smoothing, using the mean of each predefined cube region (i.e., 2x2x2) as a new voxel. Empirically, in all the tests, there was no improvement in classification accuracy and the optimal method was 'IG peeking'.

5) Feature reduction: A systematic feature reduction method can be used iteratively in the training stage to reduce the amount of voxels by an order of a magnitude, i.e, to remove redundant features, we reduce features to nearest power of two until we see a drop in accuracy on the training set. The benefits are: minimal or no loss in calculation time and similar classification accuracy on the test set, as seen in Figure 3. In terms of classification accuracy, using only 32 voxels at TR-4 are identical to using all voxels with IG weight above zero, and nearly identical for TR-3. The total amount of voxels is shared by the relevant brain regions, i.e., less voxels cover each region. Therefore, it is advisable to consider both the brain region's coverage and the calculation time in every experiment.

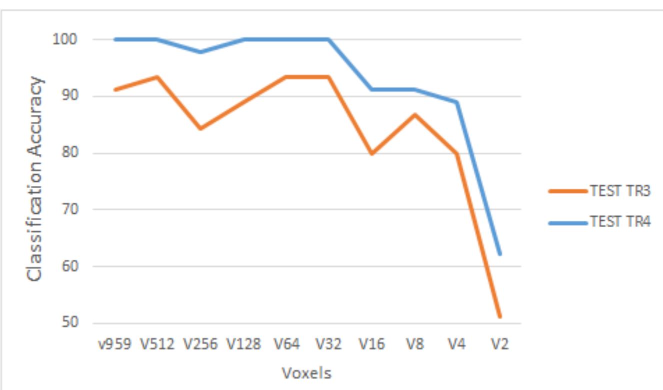

Fig. 3. Test classification accuracy in TR-3 & TR-4, for three-classes, when examining voxel amount, using a multi-class SVM classifier.

6) Classification algorithm: We used the dataset from same subject to compare several state-of-the-art algorithms such as XGboost [27] (by means of a Weka extension that wraps the MLR R-package), random forest and multi-class [28] SVM with polynomial (1st, 2nd and 3rd degrees) and RBF kernels, using a hyper-parameter grid search. Chronological run-based results, with feature selection, indicate that an SVM classifier (polynomial kernel, exponent=1.0, C=1.0, Platt's SMO version of the SVM learning algorithm [29]) outperforms XGboost's accuracy by 9%. SVM also outperforms in training time, i.e., relative to SVM's baseline training-time, XGboost's and RF's were slower by 92% and 10.5%, respectively. We chose a chronological run-based train-test split over cross-validation due to the nature of the data, the fMRI recording depicts a snapshot of the brain in time and the brain keeps changing throughout the experiment. Therefore, we can't use information from the future by using cross-validation.

7) Real-time testing: During the real-time stage we classify a data sample every TR (2 seconds) and use the same noise reduction, eye filtering and normalization methods as in the training stage, and select the same voxels based on the IG filtering performed at model training. There is a small difference between the normalization process in the testing and the training stages; in the testing stage the normalization algorithm "waits" until it accumulates enough data samples to satisfy the initial length of the sliding window, but in the training part we have the data samples beforehand, thus the normalization looks ahead instead of waiting. Finally, the data is passed into the trained SVM model, and the classification result is then transmitted to the external application. The ML-based method we have presented here is simple and computationally efficient for real-time fMRI BCI. The training process takes several minutes and the classification process takes approximately 50 milliseconds.

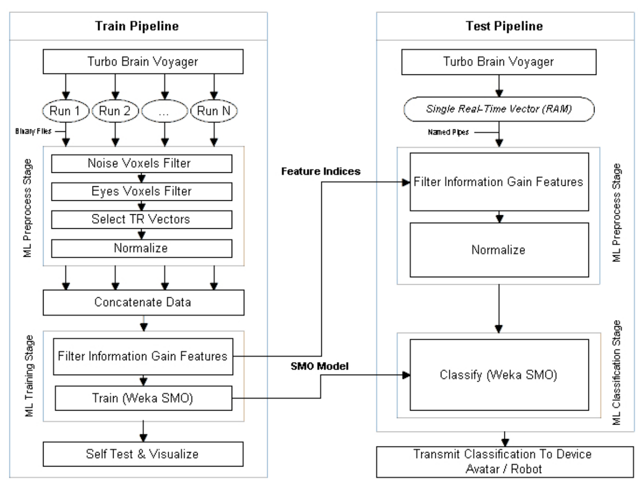

Fig. 4. The fMRI BCI ML-based system architecture. The tool is able to process both brain data arriving in real time from the fMRI scanner and pre-recorded fMRI data. The tool uses user datagram protocol (UDP) to transfer the subject's classified intentions to external devices such as an avatar or a humanoid robot.

III. RESULS: EMPIRICAL STUDIES

The system has been used in the context of several studies [4], [6], [18], [30], [31]. A typical study is divided into three parts. In the first part the subject is asked to follow a predefined protocol and several runs are recorded for the purpose of training a classifier. This is followed by the cue-based BCI part of the study, which is similar to training with the difference of providing the subject with feedback based on the classified brain activity. In the last part, the subjects perform a task, our tool classifies their intentions in real time, and the classification result is transmitted to an external device, as seen in Figure 4.

Our studies include motor execution (ME), in which the subjects are asked to move their fingers and toes, motor imagery (MI), and a visual category task (VIS, described below). The subject is provided with auditory cues in ME and MI, and in VIS the cue is the appearance of the image itself. The different classes are equally selected and triggered in a pseudo-random order in order to avoid expectation and habituation. In the ME study, the subject was expected to perform a physical action (or keep the mental imagery in his mind in the MI condition) during the trial until he heard a "rest" cue. The classification and the feedback took place 10 seconds after the voice command. We verified, by inspecting the brain activation, that the classification is based only on motor related areas, and is not based on auditory processing (e.g., responding to the auditory cues) or the eyes (e.g., looking at the direction that you expect the avatar to move to). Additionally, for the amputated population, the ME paradigm is more suitable than MI, due to their previous ability to fully control their limbs. In the MI study, we have verified that subjects were not moving their body during the experiment by constant visual inspection (through a transparent window and two video cameras). In addition, several subjects performed one session with simultaneous electromyogram (EMG) recording in order to validate that no muscle activity is involved in MI. In VIS we had subjects watch 40 blocks from four visual categories: faces, houses, tools, and a fixation screen that corresponds with idle viewing. Each block was composed of a sequence of images from one category, to elicit a strong neural response. In each block there was always one image that did not match the category, but the subject had no prior knowledge of this statistic; this was done in order to assure the subject's full attention.

In order to establish a classification performance baseline, we compared ML-based results in ME and MI to ROI-based results obtained in a previous study [6]. The ROI method was based on the Z-score formula for classification and manually localizing brain areas [18], [6], [31]. In the ML-based studies we use ML techniques to classify intentions of ME, MI and VIS. In the primary method we used all the IG voxels with a weight above 0; this is the best option for using every influential voxel detected by IG and achieving the highest possible accuracy, for example, cases where there is a small amount of voxels. In the 'voxel reduction' method we reduced processing and classification time without sacrificing classification accuracy. We suggest using the 'voxel reduction' method when the amount of voxels is high, for example, cases where there are multiple brain systems in use. This method minimizes the informative voxels and obtains similar classification results to the primary method. Our results were obtained offline from recorded data; however, the tool simulated real-time processing by sending a new brain-scan file every two seconds.

A. Offline Results

The following results are based on a 3-class task, i.e., "left", "right" and "forward", baseline of 33.3%. Figure 5 compares average ME test accuracy, over all subjects between ML and ROI. Classification accuracy coincides with the hemodynamic response: in TRs 1 and 2 the accuracy is around chance level, then it gradually increases with the best accuracy in TR 3 to TR 5-6 (6-12 seconds after the cue), and gradually drops back to chance level. The results indicate that identifying the most relevant features by using IG and classifying using SVM is superior to the ROI method. In ME above 90% average accuracy can be achieved even at TR 3 (6 seconds after a cue), which is better than the 10 seconds delay in the ROI method. By reducing the voxel count (i.e., raising the IG threshold) to the smallest number of voxels until we see a drop in accuracy on the training set, we can achieve similar accuracy to that achieved by using all voxels with positive IG, but with greatly reduced computation time. For exploration purposes we have trained classifiers using only 16 voxels, over all subjects. The average test classification accuracy was surprisingly high: 83.3%, 94% and 98.6% in TRs 3,4 and 5, respectively.

In MI (Figure 6), the average test accuracy is 90% in TR3 and 95% in TR4, which is higher than the maximum accuracy obtained with the ROI method at TR5. In VIS (Figure 7), multi-subject average results indicate an average test accuracy of 78% in TR3, and an average accuracy of up to 87% can be reached with a longer delay.

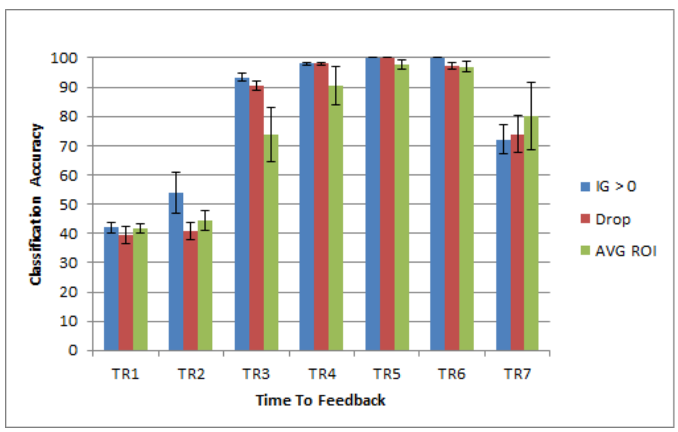

Fig. 5. A comparison of ME classification accuracy over five subjects, between ML and ROI. The ML results were obtained by using (a) all voxels with IG above 0 and (b) the smallest number of voxels until we see a drop in accuracy on the training set. Error bars indicate the 95% confidence interval.

B. Online Results

In the online experiments a "rest" command for stopping movement was introduced, i.e., a 4-class task ("left", "right",

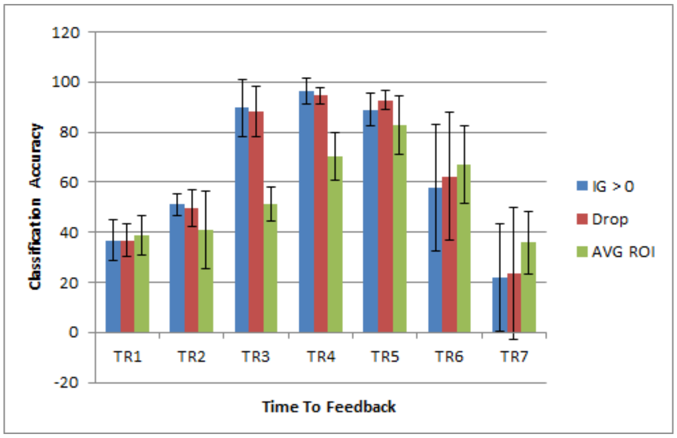

Fig. 6. A comparison of MI classification accuracy over three subjects, between ML and ROI. The ML results were obtained by using (a) all voxels with IG above 0 and (b) the smallest number of voxels until we see a drop in accuracy on the training set. Error bars indicate the 95% confidence interval.

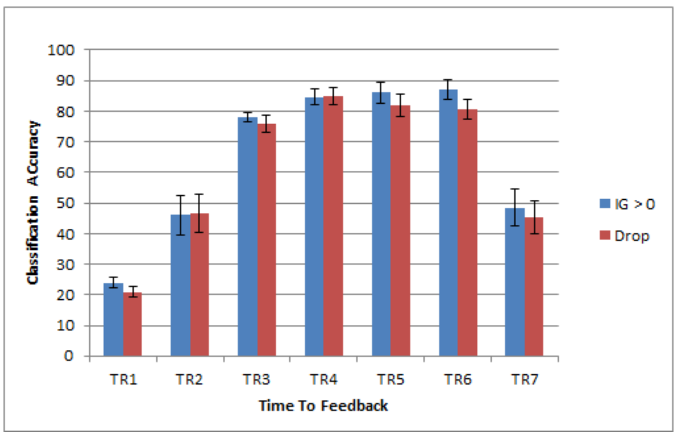

Fig. 7. ML results obtained for VIS using (a) all voxels with IG above 0 and (b) the smallest number of voxels until we see a drop in accuracy on the training set. Error bars indicate the 95% confidence interval.

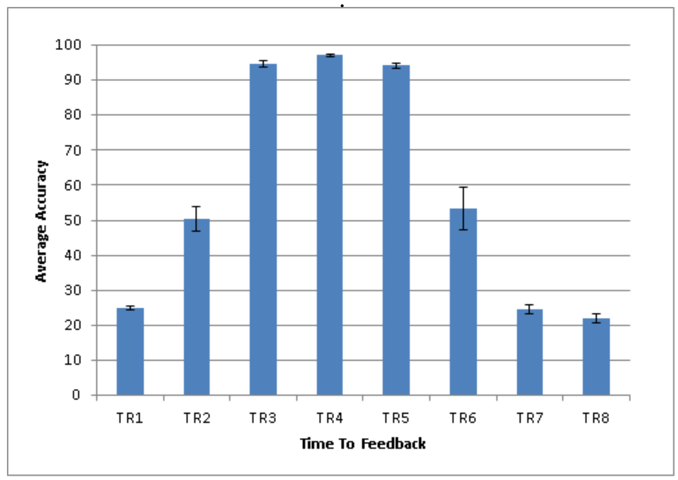

Fig. 8. Average ME classification accuracy for six subjects from a separate run from the classifier's training session. Error bars indicate the 95% confidence interval.

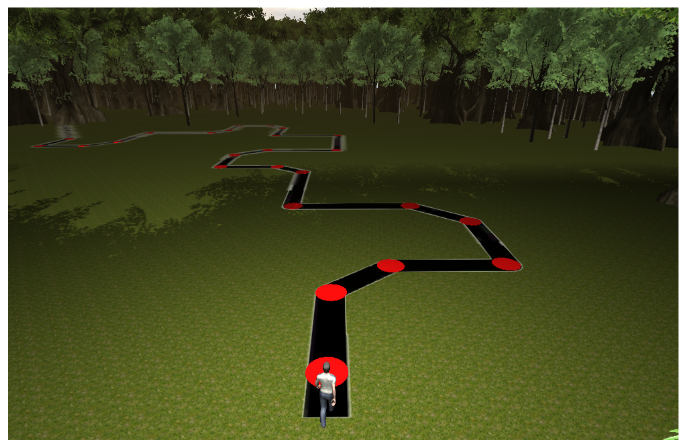

Fig. 9. The subject needs to control an avatar, navigating along the path and collecting as many red discs as possible. To successfully collect a disc, the avatar must touch it and then the disc changes to green.

"forward", and "rest"). The real-time ME experiments were conducted on several days, i.e., for every subject we train a classifier once, and continue to use it in every subsequent trial, over multiple days. The ME and MI classifiers were trained using 4 and 5-6 training runs, respectively; and were tested on a single run, averaged over all subjects, to determine the classifier's accuracy on real-time unseen data.

1) Cue-Based BCI: Figure 8 shows average classification of online accuracy over all subjects and indicates that the optimal TR in terms of accuracy is 6 seconds after a cue. The optimal classification TRs in MI are TR5 (S7, 80%) and TR3 (S8, 55%).

2) Free-Choice BCI: In the online free-choice experiment, each subject underwent between 6-15 and 8-10 free choice runs and was instructed to control an avatar and follow a predefined path while collecting as many discs as possible (Figure 9). In this stage a "rest" command for stopping movement was introduced, i.e., a 4-class task ("left", "right", "forward", and "rest"). For every subject we train a classifier once, and continue to use it in every subsequent trial, over multiple days. Classifiers cannot be used interchangeably between subjects due to anatomical differences. We provided feedback to the subject every 2 seconds and between feedback events the avatar keeps performing the last instruction.

The optimal time for completing the free-choice task was determined by a pilot study using a joystick. The task was repeated several times until it was completed without any mistakes and without using the 'rest' class to stop the avatar and was 35 seconds. In each BCI trial i, performance was calculated by dividing the trial time t_i by the amount of collected discs d_i. The ME performance of the majority of trials was near optimal. MI performance was comparatively poor; this is despite the reasonable MI offline classification accuracies; Figure 10 shows two trajectories. The best time achieved by a subject in ME and MI was 49.5 seconds and 99 seconds, i.e., an overhead of 14.5 seconds and 49.5 seconds, respectively, beyond joystick performance.

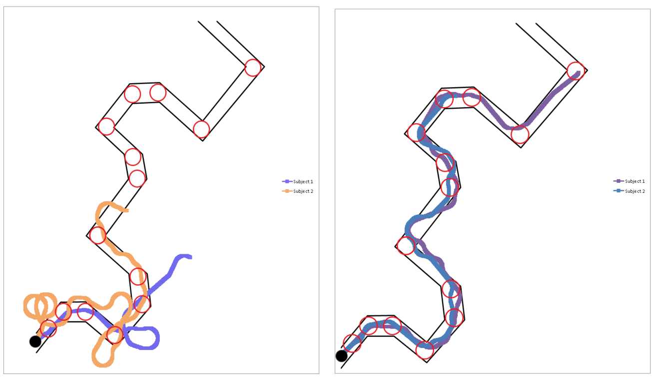

Fig. 10. A 2-Dimensional birds-eye view of optimal ME (right) and MI (left) trial performed by two subjects, the feedback was given every 2 seconds. The black spot indicates the starting point. The subjects needed to guide the avatar toward the end of the path by picking up all the red disc throughout.

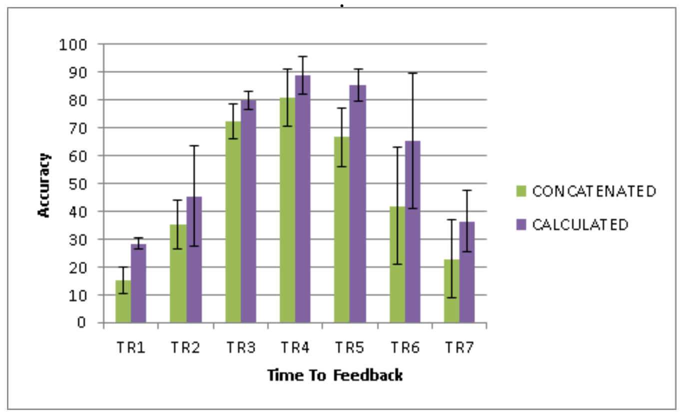

Fig. 11. A comparison of accuracy results for a 7-class dataset against a calculated average accuracy from ME and VIS. Error bars indicate the 95% confidence interval.

IV. Scalability

Our method has the advantage of being flexible. We expect it to scale to a multiple number of classes, using a wide range of mental paradigms. In order to validate our hypothesis, we created a 7-class training set, which is a concatenation of both ME and VIS conditions. The training set was composed of data recorded from three subjects who participated in both experiments. The data from the ME and VIS experiments was recorded on different days and was verified for alignment by manually inspecting the brain position and rotation of each scan in both ME and VIS; the ME data contains 4 classes and the VIS data contains 3 classes. A classifier was trained on 4 runs, using 2048 voxels selected by IG from the entire brain, which is equivalent to the sum of voxels used in the testing of both datasets separately. Testing was done on a single separated trial.

The highest classification accuracy for ME was 98.3% (4-classes, 25% baseline) and for VIS was 78.3% (3 classes, 33% baseline) at TR4. Our results indicated a 7-class accuracy at TR4 of 80.95%, as compared with the average calculated accuracy for both datasets – 88.73%, and chance accuracy of 14.2%. Figure 11 shows that the accuracy obtained from the 7-class data is very close to the accuracy calculated from both problems in TR3 and TR4, and within the confidence interval, empirically demonstrating that our tool and in turn our method can be used to create classifiers with more than 4 conditions, to classify concatenated datasets that were not recorded on the same day, to use one classifier instead of using two different classifiers, and with a minimal loss of classification accuracy.

V. Brain Analysis

Our method uses raw voxel values that cannot be compared by value to TBV's general linear model-based (GLM) voxels. However, informative and contrast-based voxels can be compared anatomically. We inspected 6 subjects that participated in cue-based MI experiments, our inspection suggests pre-motor cortex activation in MI whereas ME was mostly based on the specific body representations in primary motor cortex in all subjects. Another interesting and rather unexpected finding was that for three subjects the MI classifier identified visual cortex voxels as informative. These voxels and are pointed by the white arrows in Figure 12 were selected based on MI cue-based task, i.e., the subjects were instructed by an auditory command to do imagine an action while the avatar was idle, the selected voxels are the direct result of MI and not from a visual stimuli. This may suggest the engagement of visual imagery despite the purely motor instructions.

The validity of our MVPA method is suggested by the fact that, overall, the information gain algorithm largely selects informative voxels that are similar to those that are detected in standard GLM analysis as pointed by the blue arrows in Figure 12. The increased accuracy suggests that information gain is more sensitive than GLM.

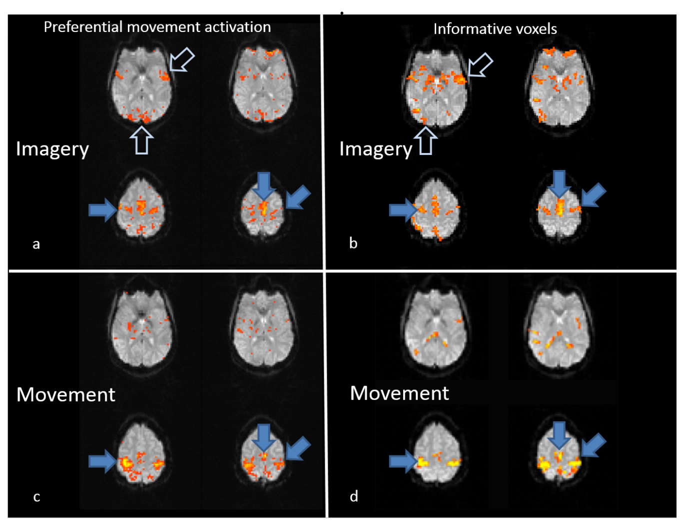

Fig. 12. A subset of corresponding slices from subject S7. The left column shows the GLM contrast $(right, left, forward) > baseline$ (thresholds: t=4.6 for ME and t=3.2 for MI), and the right column shows the 1024 voxels with highest information gain selected by our algorithm. The top row shows imagery and the bottom row shows motor execution.

VI. DISCUSSION

In this paper we demonstrated the advantage of using our tool for real-time fMRI BCI. We showed how a multivariate approach, combined with superior spatial resolution, achieves high accuracy in a series of single-trial classification and BCI studies, based on several brain networks. Our results indicate that our tool is robust and efficient in classification time, allowing for real-time fMRI experiments. Our tool can classify covert intentions that do not have specific anatomical ROIs, such as "rest", can reduce the amount data by converging only on task related voxels. The classifiers created by our tool can be used successfully, without modification, by our subjects even months later.

We showed that real-time fMRI BCI can be used with a two second delay between commands in a free choice task, and by allowing feedback that is synced with the minimal scan time, subjects are able to control an avatar without being limited to the inherent delay of the BOLD signal. After showing that fMRI can be used as a relatively fast BCI, additional mental tasks and brain activation patterns can be explored for BCI. Finally, using the tool suggested here our team and others may explore other novel BCI paradigms.

VII. ACKNOWLEDGEMENTS

This research is supported by the European Union FP7 Integrated Project VERE (No 657295), www.vereproject.eu. We would like to thank the Weizmann Institute fMRI scanner staff Edna Furman-Haran, Nachum Stern and Fanny Attar for their help and the subjects for their participation.

REFERENCES

[1] Y. Meir-Hasson, S. Kinreich, I. Podlipsky, T. Hendler, and N. Intrator, "An eeg finger-print of fMRI deep regional activation," *Neuroimage*, 2013.

[2] R. Sitaram, A. Caria, and N. Birbaumer, "Hemodynamic brain–computer interfaces for communication and rehabilitation," *Neural networks*, vol. 22, no. 9, pp. 1320–1328, 2009.

[3] J. Sulzer, S. Haller, F. Scharnowski, N. Weiskopf, N. Birbaumer, M. L. Blefari, A. Bruehl, L. Cohen, R. C. DeCharms, and R. Gassert, "Real-time fMRI neurofeedback: progress and challenges," *Neuroimage*, vol. 76, pp. 386–399, 2013.

[4] O. Cohen, D. Doron, M. Koppel, R. Malach, and D. Friedman, "High performance in brain-computer interface control of an avatar using the missing hand representation in long term amputees," in *The 8th International IEEE EMBS Conference On Neural Engineering (NER17)*, 2017.

[5] S. Kollias, X. Golay, P. Boesiger, and A. Valavanis, "Dynamic characteristics of oxygenation-sensitive mri signal in different temporal protocols for imaging human brain activity," *Neuroradiology*, vol. 42, no. 8, pp. 591–601, 2000.

[6] O. Cohen, M. Koppel, R. Malach, and D. Friedman, "Controlling an avatar by thought using real-time fMRI," *Journal of Neural Engineering*, vol. 11, no. 3, p. 035006, 2014.

[7] J. Haynes and G. Rees, "Decoding mental states from brain activity in humans," *Nature Reviews Neuroscience*, vol. 7, pp. 523–534, July 2006.

[8] K. A. Norman, S. M. Polyn, G. J. Detre, and J. V. Haxby, "Beyond mind-reading: multi-voxel pattern analysis of fMRI data," *Trends in Cognitive Science*, vol. 10, no. 9, pp. 424–430, 2006.

[9] N. Weiskopf, F. Scharnowski, R. Veit, R. Goebel, N. Birbaumer, and K. Mathiak, "Self-regulation of local brain activity using real-time functional magnetic resonance imaging (fmri)," *Journal of Physiology-Paris*, vol. 98, no. 4, pp. 357–373, 2004.

[10] S. M. LaConte, "Decoding fMRI brain states in real-time," *Neuroimage*, vol. 56, no. 2, pp. 440–454, 2011.

[11] J. R. Sato, R. Basilio, F. F. Paiva, G. J. Garrido, I. E. Bramati, P. Bado, F. Tovar-Moll, R. Zahn, and J. Moll, "Real-time fmri pattern decoding and neurofeedback using friend: an fsl-integrated bci toolbox," *PLoS One*, vol. 8, no. 12, p. e81658, 2013.

[12] R. Basilio, G. Garrido, J. Sato, S. Hoefle, B. Melo, F. Pamplona, R. Zahn, and J. Moll, "Friend engine framework: A real time neurofeedback client-server system for neuroimaging studies," *Name: Frontiers in Behavioral Neuroscience*, vol. 9, no. 3, 2015.

[13] M. Rana, N. Gupta, J. L. D. Da Rocha, S. Lee, and R. Sitaram, "A toolbox for real-time subject-independent and subject-dependent classification of brain states from fmri signals," *Frontiers in neuroscience*, vol. 7, 2013.

[14] R. Scherer, F. Lee, A. Schlögl, R. Leeb, H. Bischof, and G. Pfurtscheller, "Toward self-paced brain–computer communication: navigation through virtual worlds," *Biomedical Engineering, IEEE Transactions on*, vol. 55, no. 2, pp. 675–682, 2008.

[15] B. Sorger, J. Reithler, B. Dahmen, and R. Goebel, "A real-time fMRI-based spelling device immediately enabling robust motor-independent communication," *Current Biology*, vol. 22, no. 14, pp. 1333–1338, 2012.

[16] P. Andersson, J. P. Pluim, J. C. Siero, S. Klein, M. A. Viergever, and N. F. Ramsey, "Real-time decoding of brain responses to visuospatial attention using 7t fMRI," *PloS one*, vol. 6, no. 11, p. e27638, 2011.

[17] J.-H. Lee, J. Ryu, F. A. Jolesz, Z.-H. Cho, and S.-S. Yoo, "Brainmachine interface via real-time fMRI: preliminary study on thought-controlled robotic arm," *Neuroscience letters*, vol. 450, no. 1, pp. 1–6, 2009.

[18] O. Cohen, S. Druon, S. Lengagne, A. Mendelsohn, R. Malach, A. Kheddar, and D. Friedman, "fmri robotic embodiment: A pilot study," in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS EMBS International Conference on*, June 2012, pp. 314–319.

[19] R. Sitaram, S. Lee, S. Ruiz, M. Rana, R. Veit, and N. Birbaumer, "Real-time support vector classification and feedback of multiple emotional brain states," *Neuroimage*, vol. 56, no. 2, pp. 753–765, 2011.

[20] L. Minati, A. Nigri, C. Rosazza, and M. G. Bruzzone, "Thoughts turned into high-level commands: Proof-of-concept study of a vision-guided robot arm driven by functional MRI (fMRI) signals," *Medical Engineering & Physics*, vol. 34, no. 5, pp. 650–658, Jun. 2012.

[21] M. Chiew, S. M. LaConte, and S. J. Graham, "Investigation of fMRI neurofeedback of differential primary motor cortex activity using kinesthetic motor imagery," *NeuroImage*, vol. 61, no. 1, pp. 21–31, 2012.

[22] P. Andersson, J. P. Pluim, M. A. Viergever, and N. F. Ramsey, "Navigation of a telepresence robot via covert visuospatial attention and real-time fMRI," *Brain topography*, vol. 26, no. 1, pp. 177–185, 2013.

[23] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[24] A. van der Kouwe, S. Gicquel, G. Chen, F. Schmitt, M. Harder, D. Salat, A. Sorensen, B. Fischl, and A. Dale, "On-line automatic slice positioning and between-scan correction for brain mr protocols," in *Proceedings of the 11th Annual Meeting of ISMRM, Toronto, Canada*, vol. 11, 2003, p. 797.

[25] E. M. S. Thesen, O. Heid and L. R. Schad, "Prospective acquisition correction for head motion with image-based tracking for real-time fMRI," *Magnetic Resonance in Medicine*, vol. 44, pp. 457–465, 2000.

[26] J. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[27] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM, 2016, pp. 785–794.

[28] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," in *Advances in Neural Information Processing Systems*, I. Jordan, J. Kearns, and A. Solla, Eds., vol. 10. MIT Press, 1998.

[29] J. e. a. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," ,, 1998.

[30] O. Cohen, F. Keith, A. Kheddar, M. Koppel, R. Malach, and D. Friedman, "Real-time fMRI control of a humanoid robot using two brain networks simultaneously: A pilot study," in *7th Graz Brain-Computer Interface Conference 2017*, 2017.

[31] O. Cohen, S. Druon, S. Lengagne, A. Mendelsohn, R. Malach, A. Kheddar, and D. Friedman, "fmri-based robotic embodiment: Controlling a humanoid robot by thought using real-time fMRI," *PRESENCE: Teleoperators and Virtual Environments*, vol. 23, no. 3, pp. 229–241, 2014.