

A Kernel-Based Approach for DBS Parameter Estimation

V. Gómez-Orozco^(✉), J. Cuellar, Hernán F. García, A. Álvarez, M. Álvarez, A. Orozco, and O. Henao

Automatic Research Group, Universidad Tecnológica de Pereira, Pereira, Colombia
{vigomez,jfcuellar,hernan.garcia,andres.alvarez1,
malvarez,aaog,oscarhe}@utp.edu.co

Abstract. The volume of tissue activated (VTA) is commonly used as a tool to explain the effects of deep brain stimulation (DBS). The VTA allows visualizing the anatomically accurate reconstructions of the brain structures surrounding the DBS electrode as a 3D high-dimensional activate/non-activate image, which leads to important clinical applications, e.g., Parkinson's disease treatments. However, fixing the DBS parameters is not a straightforward task as it depends mainly on both the specialist expertise and the tissue properties. Here, we introduce a kernel-based approach to learn the DBS parameters from VTA data. Our methodology employs a kernel-based eigendecomposition from pairwise Hamming distances to extract relevant VTA patterns into a low-dimensional space. Further, DBS parameters estimation is carried out by employing a kernel-based multi-output regression and classification. The presented approach is tested under both isotropic and anisotropic conditions to validate its performance under realistic clinical environments. Obtained results show a significant reduction of the input VTA dimensionality after applying our scheme, which ensures suitable DBS parameters estimation accuracies and avoids over-fitting.

Keywords: Kernels · Volume of tissue activated · Deep brain stimulation

1 Introduction

Deep brain stimulation (DBS) is a surgical therapy used mainly for treatment of a variety of neurological disorders, in patients who do not respond well to medication. In particular, DBS consists of an electrode inserted inside neural tissue of the patient to modulate neural activity with applied electric pulses, which depend on amplitude, pulse-width, frequency, and the electrode characteristics [2]. Although the physiological mechanism of DBS still remains unclear, its clinical effectiveness is evident [5]. A measure of the effects of the DBS resides in the estimation of the volume of tissue activated (VTA), namely, the spatial

spread of direct neural response to external electrical stimulation as a 3D high-dimensional activate/non-activate image. Nonetheless, fixing the DBS parameters is not a straightforward task since it depends on both the specialist expertise and the brain tissue properties of each patient [6].

The VTA, and its visualization, jointly with reconstructions of the brain structures surrounding the implanted electrode have been proposed as an alternative to accelerate the process of stimulation parameters adjustment, also minimizing the adverse side effects that can occur when they are not carefully adjusted [1]. This system allows the medical specialist to observe the brain structures that are responding directly to the electrical stimulation. Thus, the clinician determines the possible effects of a given stimulation configuration on the patient. Nonetheless, such an approach still involves a heuristic search method (trial and error), requiring high computational load and appropriate expertise by the specialist [6]. So, the problem of computing a set of specific neuromodulation parameters given an objective VTA has been less studied in the DBS literature. In contrast, there is an extensive literature that attempts to estimate the VTA from the stimulation parameters [1, 10]. Authors in [7] introduced a machine learning system to predict the VTA from DBS parameter space. It allows the user to select a target to find the correlation between the calculated and the desired VTA. Once the correlation between the VTAs is calculated, the algorithm provides a possible configuration of neurostimulation parameters. However, the approach only operates under isotropic conditions and the system can not represent high stimulation parameter values and/or more than two active contacts. In [4] our group presented an alternative strategy for DBS parameter estimation from a previously specified VTA. We employed a framework based on support vector machines for multi-output regression and classification. However, our strategy is developed only under isotropic conditions.

In this work, we developed a kernel-based approach for DBS parameters estimation from VTA data. In this sense, our data-driven kernel-based scheme comprises mainly two stages (*i*) kernel-based principal components extraction from VTA samples, and (*ii*) DBS parameter estimation using kernel-based multi-output regression and classification. Moreover, our technique is developed under both ideal (isotropic tissue conductivities) and realistic (anisotropic tissue conductivities) assumptions. Our aim is to estimate neuromodulation parameters from the planned VTA to support DBS-based treatments. Results obtained show a remarkable reduction of the input VTA dimensionality after applying our feature extraction scheme, ensuring suitable DBS parameter estimation accuracies and avoiding over-fitting. The remainder of this paper is organized as follows: Sect. 2 describes the materials and methods of the approach introduced. Sections 3 and 4 describe the experimental set-up and the results obtained, respectively. Finally, the concluding remarks are outlined in Sect. 5.

2 Materials and Methods

Let $\mathbf{X} \in \{0, 1\}^{N \times P}$ and $\mathbf{Y} \in \mathbb{R}^{N \times Q}$ be a given pair of matrices coding the VTA and the DBS parameter spaces, respectively, holding P axons, Q

stimulation parameters, and N samples. The i -th VTA $\mathbf{x}_i \in \mathbb{R}^P$ ($i \in \{1, 2, \dots, N\}$) is computed from the DBS configuration in row vector $\mathbf{y}_i \in \mathbb{R}^Q$. In particular, a six-dimensional DBS parameter vector $\mathbf{y} = [D_A D_W c_0 c_1 c_2 c_3]$ is considered ($Q = 6$), where $D_A \in \mathbb{R}^+$ refers to the amplitude, $D_W \in \mathbb{R}^+$ to the pulse width, and c_r to the r -th contact condition, with $r = \{0, 1, 2, 3\}$. Regarding the VTA simulation, given a DBS parameter vector \mathbf{y}_i , a finite element method (FEM) is employed to compute the spatial distribution of the extracellular potential [10]. Then, a model of multicompartment myelinated axons is implemented to determine the axonal response to electric stimulation and the VTA is computed to determine the volume generated by the active axons [1]. The i -th VTA \mathbf{x}_i is stored towards axon concatenation, where the element $x_{ip} = 1$ ($p \in \{1, \dots, P\}$) if the p -th axon is activated by the DBS, otherwise, $x_{ip} = 0$. Figure 1 shows the VTA estimation sketch.

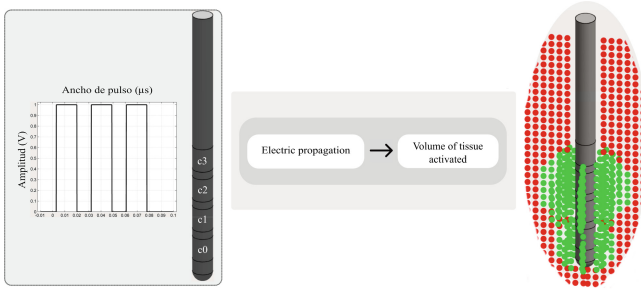


Fig. 1. VTA estimation main sketch. The VTA is composed by elements labeled as active ‘1’ (green dots) and non-active ‘0’ (red dots). (Color figure online)

With the aim to estimate the neuromodulation parameters from a planned VTA in DBS-based treatments, we introduce a data-driven kernel-based scheme to highlight the relevant relations between the VTA and the DBS parameter spaces. Our approach comprises two main stages (*i*) VTA feature extraction through kernel-based eigendecomposition, (*ii*) DBS parameter estimation using kernel-based multi-output regression and classification.

Kernel-Based VTA Feature Extraction. Let $\phi: \{0, 1\}^P \rightarrow \mathcal{H}$ be a nonlinear mapping function that embeds any $\mathbf{x} \in \mathbf{X}$ into the element $\phi(\mathbf{x}) \in \mathcal{H}$ of the Reproducing Kernel Hilbert Space (RKHS) \mathcal{H} . By assuming that the elements in \mathcal{H} are centered ($\sum_{i=1}^N \phi(\mathbf{x}_i) = 0$), the covariance matrix in the RKHS can be computed as follows: $\mathbf{S} = (1/N) \sum_{i=1}^N \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^\top$, where its eigenvalues $\lambda_m \in \mathbb{R}^+$ and eigenvectors $\mathbf{v}_m \in \mathcal{H}$ satisfy $\mathbf{S} \mathbf{v}_m = \lambda_m \mathbf{v}_m$ ($m = \{1, 2, \dots, M_P\}$, $\mathbf{v}_m \neq 0$). Since all solutions \mathbf{v}_m lie in the span of $\{\phi(\mathbf{x}_i)\}_{i=1}^N$ we may consider the equivalent system: $\lambda_m \phi(\mathbf{x}_i)^\top \mathbf{v}_m = \phi(\mathbf{x}_i)^\top \mathbf{S} \mathbf{v}_m$, and that there exists a coefficient vector set $\{\alpha_m \in \mathbb{R}^N\}_{m=1}^{M_P}$, such that $\mathbf{v}_m = \sum_{i=1}^N \alpha_i^m \phi(\mathbf{x}_i)$ for all $\alpha_i^m \in \alpha_m$. Then, an eigenvalue problem is solved to find α_m as [9]: $N \lambda_m \alpha_m = \mathbf{K} \alpha_m$, where

$\mathbf{K} \in \mathbb{R}^{N \times N}$ is a kernel matrix holding elements $k_{ij} = \kappa_x(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$, being $\kappa_x: \{0, 1\}^P \times \{0, 1\}^P \rightarrow \mathbb{R}^+$ a positive definite kernel. Due to the binary structure of the VTA, a Gaussian kernel is computed from a Hamming-based distance:

$$\kappa_x(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-d_h^2(\mathbf{x}_i, \mathbf{x}_j)}{2\sigma_x^2}\right), \quad (1)$$

where $d_h^2(\mathbf{x}_i, \mathbf{x}_j) = \sum_{p=1}^P (1 - \delta(x_{ip} - x_{jp}))$ is the Hamming distance operator, $\delta(\cdot)$ stands for the delta function, and $\sigma_x \in \mathbb{R}^+$ is kernel bandwidth. Afterwards, the kernel principal component extraction (KPCA) $z_i^m \in \mathbb{R}$ of $\phi(\mathbf{x}_i)$ onto the m -th basis in \mathcal{H} is computed as:

$$z_i^m = \mathbf{v}_m^\top \phi(\mathbf{x}_i) = \sum_{j=1}^N \alpha_j^m \kappa_x(\mathbf{x}_i, \mathbf{x}_j) \quad (2)$$

and the feature extraction matrix $\mathbf{Z} \in \mathbb{R}^{N \times M_P}$, holding row vectors $\mathbf{z}_i \in \mathbb{R}^{M_P}$, is built by concatenating the M_P principal components.

Kernel-Based DBS Parameter Estimation. Given \mathbf{Z} we build two kind of vector-valued functions. The former, $f^R: \mathbb{R}^{M_P} \rightarrow \mathbb{R}^2$ estimates the amplitude and pulse-width DBS parameter vector $\mathbf{y}_i^R \in \mathbb{R}^2$ in $\mathbf{Y}^R \in \mathbb{R}^{N \times 2}$ as: $\hat{\mathbf{y}}_i^R = f^R(\mathbf{z}_i) = \varphi^R(\mathbf{z}_i)\mathbf{W} + \mathbf{b}$, where $\mathbf{W} \in \mathbb{R}^{M_R \times 2}$, $\mathbf{b} \in \mathbb{R}^2$, and $\varphi^R: \mathbb{R}^{M_P} \rightarrow \mathbb{R}^{M_R}$. Then, a multi-output support vector regression (MSVR) optimization problem can be defined as follows [8]:

$$\mathbf{W}^*, \mathbf{b}^* = \arg \min_{\mathbf{W}, \mathbf{b}} \frac{1}{2} \sum_{m=1}^2 \|\mathbf{w}_m\|^2 + \gamma_R \sum_{i=1}^N \varsigma(u_i), \quad (3)$$

where $\mathbf{w}_m \in \mathbb{R}^{M_R}$ is the m -th column vector of \mathbf{W} , $\gamma_R \in \mathbb{R}^+$ is a regularization parameter, $u_i = (\mathbf{e}_i^\top \mathbf{e}_i)^{1/2}$, $\mathbf{e}_i = \mathbf{y}_i^R - \hat{\mathbf{y}}_i^R$, and $\varsigma(u_i) = (u_i - \epsilon)^2$ if $u_i \geq \epsilon$, otherwise, $\varsigma(u_i) = 0$ ($\epsilon \in \mathbb{R}^+$). Writing Eq. (3) in terms of $\boldsymbol{\xi}_m \in \mathbb{R}^N$, with $\boldsymbol{\varphi} = [\varphi^R(\mathbf{z}_1), \dots, \varphi^R(\mathbf{z}_N)]^\top \in \mathbb{R}^{N \times M_R}$ and $\mathbf{w}_m = \boldsymbol{\varphi}^\top \boldsymbol{\xi}_m$, a dual problem can be solved as:

$$\hat{\mathbf{y}}_i^R = \mathbf{k}_i^R \boldsymbol{\Xi} + \mathbf{b} \quad (4)$$

where $\boldsymbol{\Xi} \in \mathbb{R}^{N \times 2}$ is a weighting matrix with column vectors $\boldsymbol{\xi}_m$ and $\mathbf{k}_i^R \in \mathbb{R}^N$ is a row vector holding elements: $k_{ij}^R = \kappa_z(\mathbf{z}_i, \mathbf{z}_j)$, ($i, j \in \{1, 2, \dots, N\}$), being $\kappa_z: \mathbb{R}^{M_P} \times \mathbb{R}^{M_P} \rightarrow \mathbb{R}^+$ a Gaussian kernel function:

$$\kappa_z(\mathbf{z}_i, \mathbf{z}_j) = \exp\left(\frac{-d_e^2(\mathbf{z}_i, \mathbf{z}_j)}{2\sigma_z^2}\right); \quad (5)$$

notation $d_e(\cdot, \cdot)$ stands for the Euclidean distance and $\sigma_z \in \mathbb{R}^+$. Then, an iteratively reweighted least squares procedure is used to find $\boldsymbol{\Xi}, \mathbf{b}$ [8].

Regarding the latter function, $f^C: \mathbb{R}^{M_P} \rightarrow \{-1, 0, 1\}$, which allows computing the DBS contact configuration vector $\mathbf{y}_i^C \in \{-1, 0, 1\}^4$ in $\mathbf{Y}^C \in \{-1, 0, 1\}^{N \times 4}$, we built a soft margin support vector classifier (SVC) over \mathbf{Z} to compute the r -th contact value as:

$$\hat{y}_{ir}^C = f_r^C(\mathbf{z}_i) = \sum_{j=1}^N \varrho_j^T y_{jr}^C \kappa_z(\mathbf{z}_i, \mathbf{z}_j) + a_r, \quad (6)$$

where $g_j^r \in \mathbb{R}$ is the weight of training sample j for the r -th classifier and $a_r \in \mathbb{R}$ is a bias term. So, each classifier is solved as a quadratic optimization from the well-known SVC dual formulation (for details see [9]).

3 Experimental Set-Up

We built two VTA databases generated by 1000 randomly selected combinations of realistic stimulation parameters, thereby, $c_r \{-1, 0, 1\}$, $D_A \in [0.5, 5.5][V]$, and $D_W \in [60, 450] [\mu s]$. Such parameter value ranges are relevant in the context of VTA estimation for the Medtronic ACTIVA-RC stimulator.¹ The first database is built for both monopolar and bipolar conditions (one or two active contacts), under the assumption of an isotropic tissue medium (ITM), which is the most commonly used in clinical practice settings. The second database comprises isotropic and anisotropic tissue medium conditions (IATM), namely, 500 VTAs are computed for each of them. An extracellular potential model is executed for both databases using the COMSOL Multiphysics 4.2 FEM toolbox. Therefore, a model where the electric conductivity of the brain tissue is assumed to be homogeneous and isotropic is employed for ITM, meanwhile an anisotropic conductivity one is used for IATM. Such anisotropic conductivities are obtained from magnetic resonance imaging by means of diffusion tensors. For concrete testing, a DTI30 dataset is considered² with the RESTORE (Robust Estimation of Tensor by Outlier Rejection) algorithm, and then, linearly transformed to conductivity tensors. After that, a model of multicompartment myelinated axons is implemented by using NEURON 7.3 as a Python module, to determine axonal response to the electric stimulation. Nevertheless, solving the gold standard approach for VTA estimation is computationally expensive. So, we use a Gaussian Process classifier (GPC) to emulate the multicompartment myelinated axonal model [3]. In this sense, the multicompartment axon model is executed by a random sample set estimated from the whole axonal population, aiming to simulate the axonal response to the electric stimulation by training the GPC.

For all the experiments, we performed a training-testing validation scheme with 30 repetitions, where 80% of the samples are used as training set and the remaining 20% as testing set. Two kind of systems are tested. The former, high-dimensional kernel learning (HDKL), does not include the KPCA stage for learning relevant components from the VTAs. Instead, the MSVR and the SVC are applied directly from the input set \mathbf{X} by applying a Gaussian kernel from a Hamming-based distance. The latter, low-dimensional kernel learning (LDKL), includes the KPCA stage as feature extraction. For both systems the MSVR algorithm is implemented according to an open source code³. Furthermore, the kernel bandwidth values are fixed as: $\sigma_x = \text{med}(d_h(\mathbf{x}_i, \mathbf{x}_j))$ and $\sigma_z = \text{med}(d_e(\mathbf{z}_i, \mathbf{z}_j))$, respectively, where $\text{med}(\cdot)$ stands for the median operator. Moreover, the number of projected features in KPCA is computed from the set $M_R = \{3, 4, \dots, 30\}$.

¹ Available at www.aann.org/pdf/cpg/aanndeepbrainstimulation.pdf.

² Available at www.cabiatl.com/CABI/resources/dti-analysis/.

³ http://isp.uv.es/soft_regression.htm.

Additionally, the MSVR free parameters are fixed within the following ranges based on the system performance: $\sigma_z = [0.5\text{med}(d_e(\mathbf{z}_i, \mathbf{z}_j)), 1.5\text{med}(d_e(\mathbf{z}_i, \mathbf{z}_j))]$, $\epsilon = [0.5, 2]$, and $\gamma_R = [0.5, 2]$. Alike, the SVC regularization parameter γ_C is tuned from: $\gamma_C = [1, 10, 50, 100, 1000]$. For all the datasets provided, we uniformly subsample each VTA vector to obtain the input dimensionalities $P = \{17.405, 8.703, 3481\}$ and $P = \{17.924, 8.897, 3559\}$ in ITM and IATM, respectively.

4 Results and Discussion

Figure 2(a) and (b) show the kernel-based eigendecomposition projections for both VTA datasets studied. Each dot represents a different neurostimulation configuration, where its size is given by the contact condition, and the color represents the amplitude value. After visual inspection of the ITM results, we note that the kernel-based feature extraction allows differentiating between active and non-active points. Since ITM is computed for isotropic conditions a smooth data structure is revealed by the KPCA projection. In fact, the projected space allows coding the DBS amplitude information, which probes the capability to encode high-dimensional sample relations. Nevertheless, some overlaps are found due to different combinations of the amplitude and pulse width values, which lead to similar VTAs. Regarding the IATM visual inspection results, the achieved projection highlights more overlaps in terms of contact state and amplitude value in comparison to the ITM results. The fact that both isotropic and anisotropic conditions are studied in the same dataset leads to complex data relations between VTAs, however, a bottom to top amplitude increases is presented.

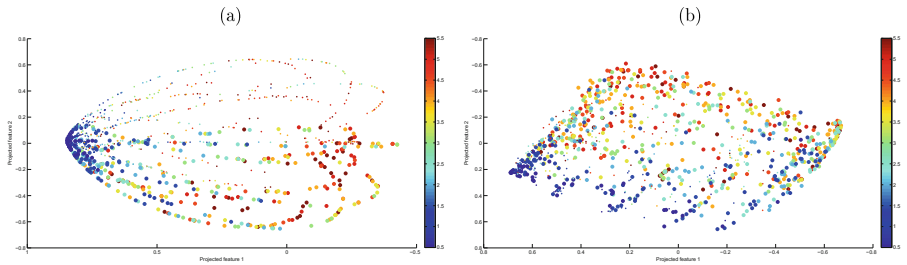


Fig. 2. Kernel-based eigendecomposition results. The dots color represents the amplitude value and the size the contact settings: active (big dot), inactive (small dot). (a) ITM dataset-Contact 0. (b) IATM dataset-Contact 0. (Color figure online)

Now, Tables 1 and 2 summarize the DBS parameter estimation accuracies obtained for both considered kernel-based approaches (HDKL and LDKL). The embedding dimensionality in LDKL is varied in the range $M_P = \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70\}$ and the best result is presented for each provided input dimensionality value. As seen, the HDKL approach

Table 1. DBS parameter estimation results for the ITM dataset. Percentage of training-set selected as support vector is shown in parentheses.

DBS par.	HDKL	LDKL ($M_P = 40$)	HDKL	LDKL ($M_P = 40$)	HDKL	LDKL ($M_P = 40$)
	$P = 17405$	$P = 17405$	$P = 8703$	$P = 8703$	$P = 3481$	$P = 3481$
D_A [V]	87.62 ± 1.43 (44.12 ± 9.93)	91.78 ± 0.84 (72.50 ± 4.41)	87.49 ± 1.26 39.42 ± 5.08	91.76 ± 0.86 (72.44 ± 3.53)	87.28 ± 1.11 44.26 ± 9.96	91.82 ± 0.88 (72.20 ± 4.50)
D_W [μ s]	84.65 ± 1.19 (44.12 ± 9.93)	85.64 ± 0.89 (72.50 ± 4.41)	84.83 ± 1.02 (39.42 ± 5.08)	85.55 ± 0.96 (72.44 ± 3.53)	84.50 ± 1.10 (44.26 ± 9.96)	85.64 ± 0.90 (72.20 ± 4.50)
c_0	97.47 ± 0.86 (21.07 ± 3.88)	97.70 ± 0.96 (13.27 ± 2.95)	97.25 ± 1.14 (22.16 ± 3.91)	95.95 ± 3.43 (14.46 ± 2.99)	97.37 ± 0.90 (20.61 ± 3.70)	97.25 ± 1.11 (13.91 ± 3.37)
c_1	94.92 ± 1.63 (23.37 ± 4.03)	97.47 ± 1.70 (15.28 ± 3.60)	94.84 ± 1.69 (20.87 ± 0.69)	96.57 ± 3.47 (15.79 ± 4.88)	94.92 ± 1.59 (23.94 ± 4.35)	97.55 ± 0.98 (14.63 ± 2.68)
c_2	95.68 ± 1.44 (23.97 ± 3.67)	98.12 ± 0.82 (14.12 ± 2.41)	95.11 ± 1.19 (25.98 ± 5.03)	96.92 ± 3.39 (15.42 ± 2.51)	95.71 ± 1.28 (24.06 ± 3.98)	96.75 ± 4.23 (16.99 ± 6.83)
c_3	97.65 ± 0.99 (21.10 ± 4.10)	98.53 ± 0.76 (13.28 ± 2.10)	96.62 ± 1.10 (20.53 ± 4.03)	96.38 ± 2.88 (13.34 ± 2.70)	97.50 ± 1.02 (20.53 ± 4.13)	97.70 ± 1.34 (12.37 ± 2.25)
<i>Average</i>	93.00 ± 1.26 (26.73 ± 5.12)	94.87 ± 0.99 (25.69 ± 3.09)	92.68 ± 1.24 (25.79 ± 3.75)	93.85 ± 2.49 (26.29 ± 3.32)	92.90 ± 1.17 (26.68 ± 5.22)	94.48 ± 1.54 (26.02 ± 3.93)

Table 2. DBS parameter estimation results for the IATM dataset. Percentage of training-set selected as support vector is shown in parentheses.

DBS par.	HDKL	LDKL ($M_P = 40$)	HDKL	LDKL ($M_P = 40$)	HDKL	LDKL ($M_P = 40$)
	$P = 17793$	$P = 17793$	$P = 8897$	$P = 8897$	$P = 3559$	$P = 3559$
D_A [V]	87.74 ± 1.14 (40.08 ± 6.22)	89.31 ± 0.91 (73.08 ± 15.38)	87.57 ± 1.14 39.92 ± 5.74	89.28 ± 0.91 (74.09 ± 15.18)	87.89 ± 1.17 (43.10 ± 9.57)	89.45 ± 0.87 (75.96 ± 14.93)
D_W [μ s]	82.91 ± 1.39 (40.08 ± 6.22)	85.13 ± 1.45 (73.08 ± 15.38)	82.95 ± 1.50 39.92 ± 5.74	85.15 ± 1.45 (74.09 ± 15.18)	82.66 ± 1.48 (43.10 ± 9.57)	85.15 ± 1.43 (75.96 ± 14.93)
c_0	86.78 ± 2.09 (38.64 ± 3.57)	84.15 ± 2.22 (33.42 ± 5.16)	86.21 ± 2.11 (38.17 ± 3.47)	83.53 ± 2.23 (34.34 ± 5.68)	85.53 ± 2.34 (38.87 ± 3.78)	83.47 ± 2.13 (33.90 ± 6.43)
c_1	80.10 ± 2.80 (51.40 ± 4.60)	77.74 ± 2.38 (43.60 ± 5.35)	79.51 ± 2.97 (52.14 ± 4.78)	77.67 ± 2.82 (41.98 ± 4.38)	79.63 ± 3.08 (50.07 ± 3.05)	76.60 ± 2.95 (44.65 ± 5.74)
c_2	77.69 ± 3.00 (54.43 ± 4.01)	77.50 ± 2.38 (48.40 ± 8.14)	77.72 ± 2.88 (54.35 ± 4.46)	77.75 ± 2.25 (46.09 ± 7.40)	77.10 ± 2.58 (54.81 ± 4.85)	76.67 ± 2.46 (47.79 ± 7.39)
c_3	82.45 ± 2.32 (43.27 ± 3.42)	82.16 ± 2.41 (37.93 ± 5.23)	82.51 ± 2.43 (43.16 ± 2.95)	82.19 ± 2.07 (38.94 ± 5.56)	82.14 ± 2.12 (44.39 ± 4.18)	82.11 ± 2.19 (40.51 ± 5.88)
<i>Average</i>	82.94 ± 2.12 (45.57 ± 4.36)	82.67 ± 1.96 (47.29 ± 7.85)	82.75 ± 2.17 (45.55 ± 4.28)	82.60 ± 1.96 (47.09 ± 7.64)	82.49 ± 2.13 (46.25 ± 5.09)	82.24 ± 2.01 (48.56 ± 8.07)

obtains slightly better results in terms of system accuracy in comparison to the LDKL methodology. However, the LDKL extracts a representation space of $M_P = 40$ for both ITM and IATM, without affecting significantly the DBS parameter estimation results. So, our kernel-based eigendecomposition is able to code a high dimensional VTA space in a few number of relevant feature. Hence, LDKL reduces the number of required support vectors by avoiding over-fitting. Overall, LDKL performance is over 94% and 82% in ITM and IATM, respectively, where the highest results are related to the c_0 and c_3 contacts and the lowest ones to c_1 and c_2 . The latter can be explained by their central position along the DBS device. So, the activation of the volume around the c_1 and c_2 positions can be affected by the activity of c_0 and c_3 . In this sense, similar VTAs can provide different DBS configurations, especially, for axons positions around the center of the stimulation device.

5 Conclusions

In this study, we proposed a novel kernel-based approach to estimate the DBS parameters from VTA data. The data-driven estimation introduced comprises two main stages: (i) kernel-based feature extraction from VTA samples, and (ii) DBS parameter estimation using kernel-based multi-output regression and classification. In this sense, we carried out a KPCA algorithm from pair-wise Hamming distances between VTAs to extract relevant features from high-dimensional and activated/non-activated-valued data. Then, a MSVR and SVC are trained in the projected space to learn the DBS configuration. The problem we describe in the paper has not been studied in deep in the literature. As we mention in the introduction, there is an extensive literature that attempts to estimate the VTA from stimulation parameters. However, attempting to solve the problem of computing a set of specific neuromodulation parameters given a desired VTA has been much less studied. The proposed approach is tested under both isotropic and anisotropic conditions to validate its performance under realistic clinical environments. According to the results achieved, a significant reduction of the VTA space is obtained based on the kernel-based eigendecomposition analysis, which avoids system over-fitting and ensures stable estimations. As future work, authors plans to develop different kernel-based eigendecomposition, besides the KPCA over the Hamming distances, aiming to enhance the system performance in challenging VTA configurations. Moreover, a pre-image extension of the introduced kernel-based extraction will be carried out for VTA reconstruction tasks.

Acknowledgments. This study was developed under grant provided by the project 111065740687 funded by Colciencias. H.F. García is funded by Colciencias under the program: Formación de alto nivel para la ciencia, la tecnología y la innovación – Convocatoria 617 de 2013.

References

1. Butson, C.R., Cooper, S.E., et al.: Patient-specific analysis of the volume of tissue activated during deep brain stimulation. *Neuroimage* **34**(2), 661–670 (2007)
2. Da Cunha, C., Boschen, S.L., et al.: Toward sophisticated basal ganglia neuromodulation: review on basal ganglia deep brain stimulation. *Neurosci. Biobehav. Rev.* **58**, 186–210 (2015)
3. De La Pava, I., Gómez, V., Álvarez, M.A., Henao, Ó.A., Daza-Santacoloma, G., Orozco, Á.A.: A Gaussian process emulator for estimating the volume of tissue activated during deep brain stimulation. In: Paredes, R., Cardoso, J.S., Pardo, X.M. (eds.) *IbPRIA 2015. LNCS*, vol. 9117, pp. 691–699. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-19390-8_77](https://doi.org/10.1007/978-3-319-19390-8_77)
4. Gómez, V., Alvarez, M.A., et al.: Estimation of the neuromodulation parameters from the planned volume of tissue activated in deep brain stimulation. *Revista Facultad de Ingeniería, Universidad de Antioquia* **79**, 9–18 (2016)
5. Grimaldi, G., Manto, M.: *Mechanisms and Emerging Therapies in Tremor Disorders*. Springer Science & Business Media, Heidelberg (2012)

6. Hariz, M.: Deep brain stimulation: new techniques. *Parkinsonism Relat. Disord.* **20**, S192–S196 (2014)
7. Lujan, J.L., Chaturvedi, A., et al.: System and method to estimate region of tissue activation, 19 November 2013, U.S. Patent No. 8,589,316
8. Sánchez-Fernández, M., de Prado-Cumplido, M., et al.: SVM multiregression for nonlinear channel estimation in multiple-input multiple-output systems. *IEEE Trans. Sig. Process.* **52**(8), 2298–2307 (2004)
9. Scholkopf, B., Smola, A.J.: *Learning with Kernels*. MIT Press, Cambridge (2002)
10. Yousif, N., Purswani, N., et al.: Evaluating the impact of the deep brain stimulation induced electric field on subthalamic neurons: a computational modelling study. *J. Neurosci. Methods* **188**(1), 105–112 (2010)