Unsupervised Data Analysis for Virus Detection with a Surface Plasmon Resonance Sensor

Dominic Siedhoff\textsuperscript{1,\,*}, Martin Strauch\textsuperscript{2,\,*}, Victoria Shpacovitch\textsuperscript{3} and Dorit Merhof\textsuperscript{2}

\textsuperscript{1} Computer Science VII, TU Dortmund University, e-mail: siedhoff@ls7.cs.tu-dortmund.de
\textsuperscript{2} Imaging and Computer Vision, RWTH Aachen University
e-mail: martin.strauch@lfb.rwth-aachen.de, dorit.merhof@lfb.rwth-aachen.de
\textsuperscript{3} Leibniz-Institut für Analytische Wissenschaften (ISAS)
e-mail: viktoria.shpacovitch@isas.de
\* equal contribution

Abstract—We propose an unsupervised approach for virus detection with a biosensor based on surface plasmon resonance. A column-based non-negative matrix factorisation (NNCX) serves to select virus candidate time series from the spatio-temporal data. The candidates are then separated into true virus adhesions and false positive NNCX responses by fitting a constrained virus model function. In the evaluation on ground truth data, our unsupervised approach compares favourably to a previously published supervised approach that requires more parameters.

Keywords—Unsupervised, Detection, Classification, Matrix factorisation, NNCX, Biosensor, Biological viruses

I. INTRODUCTION

Virus detection is a crucial task in health care, biological and pharmaceutical research. The Plasmon-Assisted Microscopy of Nano-Objects (PAMONO) sensor technique \cite{1} detects viruses through the surface plasmon resonance (SPR) effects they cause when binding to antibodies attached to the PAMONO sensor surface. The technique bridges diagnostic gaps as it detects the viruses causing an infection instead of the antibodies produced in response to it. It also allows to investigate the ability of antibodies to bind certain viruses, which has applications in testing newly developed antibodies in pharmaceutical research.

The PAMONO sensor records time series of 2D images, resulting in a large spatio-temporal volume of intensities, e.g. 4100 images with 750 × 230 pixels in one measurement. Hence, it is desirable to develop automated methods that can detect the virus-related SPR effects and count viruses. This involves finding contiguous areas exhibiting sudden intensity steps, which are indicative of virus adhesions (cf. virus time series in Fig. 1). These true virus adhesions must be separated from sensing artefacts and background noise (cf. also Fig. 1).

A. Related Work

The current state of the art in PAMONO data analysis are supervised approaches \cite{2, 3, 4} that separate the task into a detection stage aimed at high Recall and a classification stage aimed at high Recall and Precision: the detector generates a (possibly large) number of object hypotheses in order to find all viruses, and the classifier sorts out false positive detector responses that are due to artefacts or background noise. The approach in \cite{2} uses GPGPU for fast processing and is real-time capable on embedded systems \cite{5}.

Supervised approaches optimise detector parameters and train a classifier model for a given sensor setup and virus type. Currently, optimisation and training need to be repeated in case of larger changes to the sensor setup, which is a time-consuming offline step. In contrast, unsupervised approaches need no optimisation and training. The convex cone algorithm \cite{6} used for NNCX matrix factorisation \cite{7} is an unsupervised method that was recently applied successfully to a similar task: analysing calcium imaging videos of honeybee brain activity involves the separation of similar step signals from artefacts and noise (cf. Fig. 8 in \cite{8}). Smoothness-constrained convex cone \cite{9} is an extension that aims particularly at spatially contiguous regions of signal time series.

In the following, we utilize this algorithm as an unsupervised virus detector and develop a complementary unsupervised classifier (Section II). We evaluate both on ground truth data (Section III) and compare the results to those obtained with the supervised approach described in \cite{2} [chap. 3].

II. METHODS

A. Detector: Finding Virus Candidate Time Series with NNCX

PAMONO sensor data can be regarded as a matrix \( A \in \mathbb{R}^{t \times p} \) with \( t \) time points and \( p \) pixels. The Non-Negative CX problem (NNCX) \cite{7, 6} consists of selecting \( c \) columns of \( A \) into \( C \in \mathbb{R}^{t \times c} \) such that \( \|A - CX\|_{F^2} \) is minimised in the Frobenius norm, where the coefficient matrix \( X \in \mathbb{R}^{c \times p} \) is required to be non-negative. Optimisation of this criterion leads to a comparably small number of pixel time series in \( C \) that can reconstruct a comparably large amount of the norm of \( A \). The pixel time series in \( C \) can ‘explain’ a usually limited number of further pixels marked by the coefficients greater than zero in \( X \).

For optimisation, we employed the ConvexCone algorithm that has already been applied for NNCX on calcium imaging data of brain activity \cite{6, 8, 9}, a data type with similar properties. Briefly, the columns in \( C \) span a convex cone, the set of all linear combinations of columns \( C \) with non-negative
coefficients. As the error \( \| A - CX \|_F \) depends on the data points that are left outside of the convex cone, the ConvexCone algorithm aims to select extreme columns from \( A \) in order to span a convex cone that contains most of the data points.

In particular, we used the smoothness-constrained variant of ConvexCone [9] that allows only smooth clusters of non-zero coefficients in \( X \), thereby explicitly identifying spatially coherent clusters of pixel time series with the same underlying signal. For illustration, we performed NNCX factorisation of an example data set, obtaining virus candidate time series in \( C \) and the corresponding virus candidate locations in \( X \). Fig. 2 shows a cluster map based on the convex hull of the non-zero coefficients in each row of \( X \). In postprocessing, we replaced \( C \) by \( C := AX^\dagger(t) \) (pseudoinverse), essentially averaging time series in coherent clusters.

In order to temporally localise virus docking events, we estimated the time point of virus docking in each candidate time series as the time point of largest signal change. Signal change was computed as the difference between the means of two consecutive time windows with length \( w \).

**B. Classifier: Fitting a Virus Docking Model to the Time Series**

A virus docking event is modelled by a function \( f(x, \ldots) \) with three segments, namely the pre-docking segment \( b_1 \) (low-level plateau or drift), the virus docking segment \( b_2 \) (steep rise) and the post-docking segment \( b_3 \) (high-level plateau), cf. Fig. 1, right. \( b_1 \) is represented by a line with slope \( s_1^{\text{slope}} \). \( b_2 \) is a line starting at time point \( b_2^{\text{start}} \) that has slope \( s_2^{\text{slope}} \) and increases the signal level by \( s_2^{\text{rise}} \) units. \( b_3 \) is represented by a constant segment at signal level \( s_3^{\text{level}} \):

\[
y := f(x, s_1^{\text{slope}}, b_2^{\text{start}}, s_2^{\text{slope}}, s_2^{\text{rise}}, s_3^{\text{level}}).
\]

For time points \( x = 1, \ldots, N \), the virus docking model is defined as

\[
f(x, \ldots) := \begin{cases} s_1^{\text{offset}} + x s_1^{\text{slope}} & : 1 \leq x < b_2^{\text{start}} \\ s_2^{\text{offset}} + (x - b_2^{\text{start}}) s_2^{\text{slope}} & : b_2^{\text{start}} \leq x < b_3^{\text{start}} \\ s_3^{\text{level}} & : b_3^{\text{start}} \leq x \leq N 
\end{cases}
\]

Equation 2 contains the following dependent variables:

\[
s_2^{\text{offset}} = s_3^{\text{level}} - s_2^{\text{rise}}, \quad s_2^{\text{offset}} = s_2^{\text{offset}} - s_1^{\text{slope}} (s_2^{\text{start}} - 1) \quad \text{and} \quad s_3^{\text{start}} = (s_3^{\text{level}} - s_1^{\text{offset}})/(s_2^{\text{slope}} + s_2^{\text{start}}).
\]

We fit \( f(x, \ldots) \) to the candidate time series with an adaptive non-linear least squares algorithm (R Stats package based on [10]). Variables \( s_2^{\text{slope}} > 0 \) and \( s_2^{\text{rise}} < 0 \) (free parameter) were constrained to ensure that the step segment \( b_2 \) has a positive slope and increases the signal by at least \( \rho \) units.

If a candidate time series exhibits the ascending step shape of a prototypical virus docking event (cf. virus time series in Fig. 1), function \( f(x) \) can be fit successfully (Fig. 3a), whereas no fit can be obtained in case of strong deviations from the prototypical shape (Fig. 3b).

For classification, we compute the Pearson correlation between a virus candidate time series and the fit virus docking model \( f(x) \) with constraints on \( s_2^{\text{slope}} \) and \( s_2^{\text{rise}} \). If a fit fails due to violated constraints, the correlation is set to zero. All virus candidate time series with model correlations \( \tau \) (free parameter) are classified as virus time series. The correlation measure serves to distinguish true virus docking events (Fig. 3c: panel 1, 2) from sensor artefacts (Fig. 3c: panel 3).

**C. Postprocessing to Remove Multi-Detections**

Multiple detections of the same virus occur e.g. if the intensity increase caused by the virus is spatially fragmented by noise or artefacts. This distorts reported virus counts and is alleviated by the following heuristic, exploiting proximity between first-time and repeated detections: among the polygons output by the detector (cf. Fig. 2), those classified as viruses are sorted by descending polygon area and then spatiotemporally clustered using a cylindrical kernel: a cylinder with spatial radius \( r \) and temporal height \( h \) is placed in the centroid of the largest virus polygon. Then all other polygons within the cylinder are removed, i.e. the cluster is represented by its largest virus polygon. This process is iterated with the next smaller virus polygon until all virus polygons have been processed. A fixed choice of \( r = 20 \) pixels and \( h = \pm 25 \) frames was taken empirically by examining clusters of multi-detections.
Fig. 2. Cluster map based on an NNCX factorisation with \( c = 500 \). Positions of the selected pixel time series \( C_i^{(1)} \) in the image plane are marked by dots. The red regions around the dots contain the non-zero pixel coefficients in the corresponding rows \( X_i \).

![Image](image.png)

**Fig. 3.** Time (x-axis) vs. signal intensity (y-axis, arbitrary units) (a) Successful fits of the virus docking model. (b) Failed fits. (c) Successful fits on a data set with high SNR but a large number of sensor artefacts. Panel 1-2: true docking events. Panel 3: sensor artefact time series.

**Table 1** shows performance measures obtained using the final parameters, chosen as to be presented in Section III-B. Six PAMONO experiments obtained under different recording conditions are examined. Note that for safety reasons, polystyrene particles were used as surrogates for real viruses, hence this evaluation will use the term ‘particle’ instead of ‘virus’. Experiment names are composed of particle size and a suffix indicating either sensor surface quality (High/Medium/Low Quality), or use of a different camera (Guppy PRO F-503B). For each experiment, its median time series \( \text{SNR} \) over all particles is reported. Note that the minimum \( \text{SNR} \) in 200 nm HQ is 1.75 while that in 200 nm MQ is 1.37, hence the latter is denoted medium quality (MQ). Changing the camera in 200 nm Gpy results in higher \( \text{SNR} \) but also more artefacts (not tabulated). The 100 nm experiments exhibit lower \( \text{SNR}^{\text{med}} \) due to a linear relationship between particle diameter and signal intensity, cf. Fig. 4c in [1]. Detailed characteristics of the experiments are provided in [2] (Ch. 7). All further columns contain two rows per experiment: the upper row contains results of the unsupervised approach presented here, which are contrasted with the supervised results from [2] in the lower row. The latter are the mean values observed over three-fold cross-validation of applying the SynOpSis method, which uses image synthesis to create ground truth for optimising detector parameters and classifying model applied in a subsequent analysis (Ch. 3 in [2]).

Detector Precision is reported as \( \text{Precision}^\text{D} \) and serves as a measure of class balance for the classifier since every false positive (FP) of the detector must be sorted out by the classifier. It is not a measure of analysis quality because overall Precision is to be ensured by the classifier, and in NNCX it depends directly on the number \( c \) of extracted time series. The next column relating to the detector is \( \text{Recall}^\text{D} \), measuring the ratio of ground truth particles that could be found. \( \text{M-Rate} \) measures the relative amount of detector true positives (TPs) that are due to multi-detections of the same ground truth.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>( \text{SNR}^{\text{med}} )</th>
<th>( \text{Precision}^\text{D} )</th>
<th>( \text{Recall}^\text{D} )</th>
<th>( \text{M-Rate} )</th>
<th>( \text{Recall}^\text{D} )</th>
<th>( \text{D-Rate} )</th>
<th>\text{Specificity}</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 nm HQ</td>
<td>2.20</td>
<td>0.91</td>
<td>0.90</td>
<td>0.02</td>
<td>0.96</td>
<td>0.07</td>
<td>0.87</td>
<td>unsupervised</td>
</tr>
<tr>
<td>200 nm MQ</td>
<td>2.50</td>
<td>0.81</td>
<td>0.80</td>
<td>0.04</td>
<td>0.92</td>
<td>1.00</td>
<td>0.94</td>
<td>supervised</td>
</tr>
<tr>
<td>200 nm LQ</td>
<td>2.13</td>
<td>0.80</td>
<td>0.78</td>
<td>0.05</td>
<td>0.91</td>
<td>0.97</td>
<td>0.09</td>
<td>unsupervised</td>
</tr>
<tr>
<td>200 nm Gpy</td>
<td>3.77</td>
<td>0.81</td>
<td>0.87</td>
<td>0.07</td>
<td>0.85</td>
<td>0.96</td>
<td>0.04</td>
<td>supervised</td>
</tr>
<tr>
<td>100 nm HQ</td>
<td>1.83</td>
<td>0.58</td>
<td>0.62</td>
<td>0.03</td>
<td>0.67</td>
<td>0.97</td>
<td>0.14</td>
<td>unsupervised</td>
</tr>
<tr>
<td>100 nm LQ</td>
<td>1.25</td>
<td>0.62</td>
<td>0.71</td>
<td>0.03</td>
<td>0.87</td>
<td>0.96</td>
<td>0.24</td>
<td>supervised</td>
</tr>
</tbody>
</table>

Table 1. Median SNRs and performance measures for our unsupervised and the supervised method from [2]. Green: unsupervised is better than [2]; red: unsupervised is worse; blue: tie. Specificity is measured on virus-free control data recorded before each experiment. Details: Section III-A.

### III. RESULTS

#### A. Unsupervised vs. Supervised Approach and Ground Truth

Table 1 shows performance measures obtained using the final parameters, chosen as to be presented in Section III-B. Six PAMONO experiments obtained under different recording conditions are examined. Note that for safety reasons, polystyrene particles were used as surrogates for real viruses, hence this evaluation will use the term ‘particle’ instead of ‘virus’. Experiment names are composed of particle size in nm and a suffix indicating either sensor surface quality (High/Medium/Low Quality), or use of a different camera (Guppy PRO F-503B). For each experiment, its median time series SNR over all particles is reported. Note that the minimum SNR in 200 nm HQ is 1.75 while that in 200 nm MQ is 1.37, hence the latter is denoted medium quality (MQ). Changing the camera in 200 nm Gpy results in higher SNR but also more artefacts (not tabulated). The 100 nm experiments exhibit lower SNR\text{med} due to a linear relationship between particle diameter and signal intensity, cf. Fig. 4c in [1]. Detailed characteristics of the experiments are provided in [2] (Ch. 7). All further columns contain two rows per experiment: the upper row contains results of the unsupervised approach presented here, which are contrasted with the supervised results from [2] in the lower row. The latter are the mean values observed over three-fold cross-validation of applying the SynOpSis method, which uses image synthesis to create ground truth for optimising detector parameters and classifying model applied in a subsequent analysis (Ch. 3 in [2]).

Detector Precision is reported as Precision\text{D} and serves as a measure of class balance for the classifier since every false positive (FP) of the detector must be sorted out by the classifier. It is not a measure of analysis quality because overall Precision is to be ensured by the classifier, and in NNCX it depends directly on the number c of extracted time series. The next column relating to the detector is Recall\text{D}, measuring the ratio of ground truth particles that could be found. M-Rate measures the relative amount of detector true positives (TPs) that are due to multi-detections of the same ground truth.
NNCX \[9\] requires an additional parameter \( \lambda \) generously larger than the expected particle count. proved to be a solid choice, i.e. we recommend setting \( c \)

Across the examined experiments with 56-352 particles, could handle higher \( c \) risk of decreasing classifier Precision

B. Parameter Choice

NNCX, and a variable for the supervised approach. \( c \) detector responses, i.e. the constant parameter

negative examples. For particle-free data, \( N \)

\( (1 - C)^{+} \) indicates underestimation, making D-Rate a summary measure of the ground truth \( C \) and Recall \( C \) and Recall \( C \) of plotting Precision

Particle sample time series in the positive quadrant can be interpreted as true docking events if they are sufficiently different from the control cluster (black), which represents sensor data produced in the absence of particles. Particle-free time series are also present within the red cluster because it contains all \( c = 500 \) candidates. Time series in the lower negative quadrant are those with signal decreases in the middle segment \( s_2 \) that is characteristic for particle docking events (Fig. 1).

Particle sample time series in the positive quadrant can be interpreted as true docking events if they are sufficiently different from the control cluster (black), which represents sensor data produced in the absence of particles. Particle-free time series are also present within the red cluster because it contains all \( c = 500 \) candidates. Time series in the lower negative quadrant are those with signal decreases in the middle segment \( s_2 \) that is characteristic for particle docking events (Fig. 1).

Finally, Specificity control is defined with respect to particle-free data recorded before the respective experiment. Here, every positive classifier response is an FP\( C \). Specificity is defined as \( (1 - \text{FP}\_C) / N \), where \( N \) is the number of actually negative examples. For particle-free data, \( N \) is the number of detector responses, i.e. the constant parameter \( c \) in case of NNCX, and a variable for the supervised approach.

B. Parameter Choice

1) Detector: Parameters \( c \) and \( \lambda \) of NNCX: The only parameter for the standard NNCX factorisation [7] is the number of columns (particle candidate time series) which we set to \( c = 500 \) for all data sets. Fig. 4a shows that detector Recall\( D \) saturates around \( c = 500 \), Increasing \( c \) bears the risk of decreasing classifier Precision\( C \). However, the classifier could handle higher \( c \) in all but one case, cf. Fig. 4b. Similarly, D-Rate in Fig. 4c saturates for \( c \) around 500, except for the data set with decreasing Precision\( C \). Increasing \( c \) beyond 500 only marginally improves underestimated particle counts. Across the examined experiments with 56-352 particles, \( c = 500 \) proved to be a solid choice, i.e. we recommend setting \( c \) generously larger than the expected particle count.

The smoothness-constrained convex cone algorithm to solve NNCX [9] requires an additional parameter \( \lambda \) that influences the size of the spatially coherent clusters around the pivot pixel \( C(i) \) (cf. Fig. 2). We used the default choice from [9]: \( \lambda = 0.5 \max(X(i)) \).

We also computed the norm reconstruction \( (100 - 100 \|A - CX\|_F^2 / \|A\|_F^2) \) that could be achieved with \( c = 100, \ldots, 500 \) columns from \( A \). As the virus docking events have small spatial extension, only few neighbouring time series are correlated, which prevents a low-rank approximation based on a small number of sparse factors. However, a relatively small \( c \) can reconstruct a relatively large amount of the norm: \([c; c \text{ in } \% \text{ of the number of all columns } p; \text{ norm reconstruction (in } \%)):\ ([100; 0.08\%; 3.45\%], [200; 0.17\%; 5.51\%], [300; 0.25\%; 7.47\%], [400; 0.34\%; 9.41\%], [500; 0.42\%; 11.31\%]).

2) Classifier: Model Parameter \( \rho \) and Correlation Threshold \( \tau \): The left part of Fig. 5 shows the values of parameters \( s^\text{down}_2 \) and \( s^\text{up}_2 \) after fitting unconstrained \( f(x) \) for two exemplary data sets, recorded with different cameras. We fit the function to 500 candidate time series from either a particle sample measurement (red dots) or from the corresponding particle-free control measurement (black crosses). While control time series cluster mostly around the origin, many particle sample time series occur in the positive quadrant, indicating the intensity increase in the middle segment \( s_2 \) that is characteristic for particle docking events (Fig. 1).

Particle sample time series in the positive quadrant can be interpreted as true docking events if they are sufficiently different from the control cluster (black), which represents sensor data produced in the absence of particles. Particle-free time series are also present within the red cluster because it contains all \( c = 500 \) candidates. Time series in the lower negative quadrant are those with signal decreases in the middle segment \( s_2 \). They are caused by the intensity decreases observed around particle adhesions (Fig. 1, centre).

Based on this exploratory data analysis with an unconstrained function, we determined the constraints for the particle docking model function such that data points from the positive quadrant are selected, and such that the red dots (particle sample measurement) are well separated from the black crosses (empty control): \( s^\text{down}_2 > 0 \) and \( s^\text{up}_2 > \rho = 0.3 \) (for [0,1] normalized time series). The same constraints were used for all data sets in Table 1.

The last free parameter of the classifier is the threshold \( \tau \). Another exploratory data analysis was conducted in terms of plotting Precision\( C \) and Recall\( C \) over a histogram of the model correlations from Section II-B, cf. centre part of Fig. 5. We chose threshold \( \tau = 0.5 \) as it separates the particles which exhibit correlations close to one from artefacts and noise which exhibit correlations close to zero. Negative correlations were mapped to zero because the downward step functions arising in the vicinity of adhesions would otherwise increase the amount of multi-detections. The right part of Fig. 5 shows classifier ROC curves, each with the ROC point corresponding to \( \tau = 0.5 \). This choice prefers Recall\( C \), or equivalently true positive rate (TPR), over false positive rate (FPR).
IV. DISCUSSION

A. Selecting Particle Candidates with NNCX

NNCX has so far been employed on tabular data [7] and on calcium imaging videos of brain activity [6], [9], [8]. For the calcium imaging data, the redundancy due to large pixel clusters with correlated signals allows for norm reconstructions in the 80-90% range with \( c = 30-50 \). The columns of \( A \) are pixel time series from a brain activity video, the time series are affected by measurement noise, and time series from the same neural unit occur in large, spatially coherent clusters.

This scenario is similar to the plasmon resonance sensor recordings. Particle docking events, however, result in rather small, spatially coherent clusters (Fig. 2) of correlated time series that also have a lower SNR than the calcium imaging videos. Thus, there are few spatially localised signals immersed in a pool of noisy time series, which explains the low norm reconstructions obtained (Section III-B). However, these localised signals exhibit ascending step shapes that contribute to the norm (Fig. 3), which is the basis for successful particle detection (high detector Recall\(^D\) in Table 1) with the norm-reconstructing NNCX.
B. Performance: Unsupervised vs. Supervised Approach

Both, our unsupervised and the supervised approach [2], consist of a detector stage and a subsequent classifier stage aimed at discarding false positives.

1) Detector Stage: NNCX: Particles can be reliably detected with NNCX as shown by high Recall\(^D\), measuring the ratio of ground truth particle docking events that could be detected (Fig. 4a). Comparing the unsupervised NNCX detector to the supervised one results in a tie in terms of Recall\(^D\) in Table 1, while Precision\(^D\) only measures class balance and directly depends on \(c\).

2) Classifier Stage: Model Function Fitting: Compared to the supervised method, unsupervised function fitting achieves a slightly higher Precision at the cost of reduced Recall (cf. Precision\(^C\), Recall\(^C\) in Table 1), which remains, however, still in the 0.7-0.96 range for all but one data set (100 nm LQ).

The limitations of the model function (and correlation with the model function) become apparent in the following two cases: 1) 100 nm LQ exhibits low SNR and contains a number of particle time series with \(s_{\text{LQ}}^3 < \rho < 0.3\), i.e. the chosen constraint for function fitting is too strict, which explains the low Recall\(^C\) in this case. 2) 200 nm Gpy has a high SNR but contains a large number of artefacts, rendering it the only data set where the unsupervised function fitting has a low Precision\(^C\) (Table 1). The latter is due to unspecific fits to the artefact time series (cf. Fig. 3c). The supervised approach trades higher complexity of its trained models for a better separation of these artefacts.

3) Detector and Classifier Stage: Particle Counting: Both methods have high control Specificity (Table 1), i.e. there are virtually no particle detections in empty control measurements for either method. D-Rate, which measures the relative deviation of the estimated number of particles from the true number, differs between both methods (Table 1) but no method is generally better than the other (unsupervised: two times better, three times worse, one tie).

For the unsupervised method, large deviations from the true number of particles occurred on the two data sets discussed above where the low Precision\(^C\) for 200 nm Gpy caused a significant overestimation (D-Rate: 0.35), while the low Recall\(^C\) for 100 nm LQ caused a significant underestimation (D-Rate: -0.39) of the true number of particles.

The supervised method [2] requires 28 detector and three classifier parameters. In contrast, our unsupervised approach requires only three tunable parameters: 1) the number of columns \(c\), 2) \(\rho\) for constraining \(s_{\text{LQ}}^2\), and 3) the threshold \(\tau\) for the correlation with the model function. As shown in Section III, these parameters can be set to the same values \((c = 500, \rho = 0.3\) and \(\tau = 0.5\)) for all data sets. Furthermore, they are intuitive for wet lab technicians; \(\rho\) directly relates to signal characteristics and \(c\) to particle load and sensor size.

Note that there are additional parameters which we deem to be uncritical: 1) the constraint \(s_{\text{LQ}}^{\text{step}} > 0\) only requires that a particle signal must contain an ascending step; 2) we used the standard setting from [9] for the smoothness-parameter \(\lambda\) in the smoothness-constrained convex cone algorithm; 3) the postprocessing step (Section II-C) requires a spatio-temporal search cylinder with height \(h\) and radius \(r\); 4) estimating the time point of particle docking requires specifying length \(w\) of the time window used for averaging (Section II-A).

V. CONCLUSIONS

In this work, we have explored the possibility of virus (particle) detection with a PAMONO biosensor using unsupervised data analysis. The unsupervised NNCX factorisation as a detector of particle candidates proves to be comparable to the previously published supervised detector, while requiring no training data and fewer parameters. Despite some limitations of the function fitting at the classifier stage, e.g. in the presence of artefact time series, a completely unsupervised approach (1. NNCX detection; 2. inspection of rise vs. slope of the unconstrained functions: Fig. 5, left) allows to distinguish experiments with particles from particle-free controls. Our results indicate that unsupervised virus detection based on fast matrix algorithms is a promising avenue for further research.

ACKNOWLEDGMENT

Part of this work has been supported by Deutsche Forschungsgemeinschaft (DFG) through grant ME3737/3-1 (RWTH Aachen) and within the Collaborative Research Centre SFB 876: http://sfb876.tu-dortmund.de/

REFERENCES


