



Original software publication

NANOPTICS: In-depth analysis of NANomaterials for OPTICal localized surface plasmon resonance Sensing



Marco S. Rodrigues*, Rui M.S. Pereira, Mikhail I. Vasilevskiy, Joel Borges, Filipe Vaz

Centro de Física, Universidade do Minho, Campus de Gualtar, Braga, Portugal

ARTICLE INFO

Article history:

Received 28 March 2020

Received in revised form 20 May 2020

Accepted 20 May 2020

Keywords:

Nnanoplasmonic thin films

Optical sensing

LSPR band analysis

ABSTRACT

Thin films containing metal (gold, silver, etc.) nanoparticles are important platforms for molecular sensing based on the localized surface plasmon resonance (LSPR) phenomenon. To support their use in high-resolution LSPR spectroscopy systems, a software package, called NANOPTICS, was developed. It performs a statistical analysis of the LSPR band of the films, employing central moments of the normalized spectral distribution of optical transmittance. LSPR band parameters, signal-to-noise ratio and refractive index sensitivity are the outputs. It is shown that NANOPTICS can be a powerful tool to perform fast diagnostics of LSPR-based sensors' sensitivity and to analyse their response to target analytes.

© 2020 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Code metadata

Current code version	2020a (2.4)
Permanent link to code/repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX_2020_42
Code Ocean compute capsule	
Legal Code Licence	MIT
Code versioning system used	none
Software code languages, tools, and services used	MATLAB
Compilation requirements, operating environments & dependencies	MATLAB R2018a — App Designer
If available Link to developer documentation/manual	https://github.com/marcoprs2020/nanoptics
Support email for questions	mprodrigues@fisica.uminho.pt ; marcoprs@gmail.com

Software metadata

Current software version	2020a (2.4)
Permanent link to executables of this version	https://github.com/marcoprs2020/nanoptics
Legal Software Licence	MIT
Computing platforms/Operating Systems	Microsoft Windows 64 bit
Installation requirements & dependencies	MATLAB Runtime R2018a
If available, link to user manual—if formally published include a reference to the publication in the reference list	https://github.com/marcoprs2020/nanoptics
Support email for questions	mprodrigues@fisica.uminho.pt ; marcoprs@gmail.com

1. Motivation and significance

Chemical and physical thin film sensors have undergone several changes since their research boom in the 80's [1,2]. Since then, the growing requirements for high sensitivity, selectivity,

* Corresponding author.

E-mail address: marcoprs@gmail.com (M.S. Rodrigues).

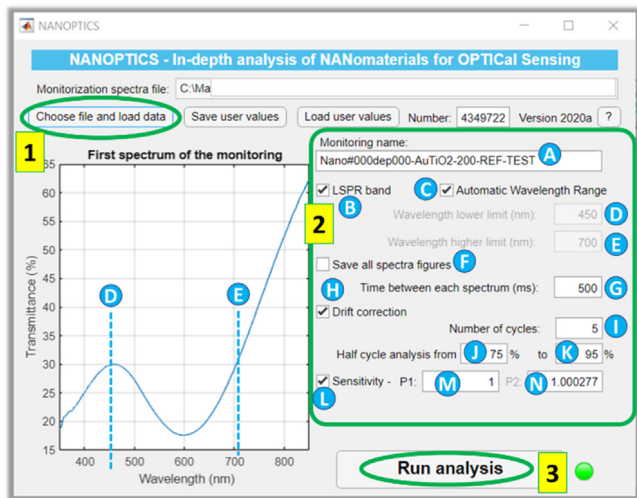


Fig. 1. GUI for spectrum visualization and setting initial parameters.

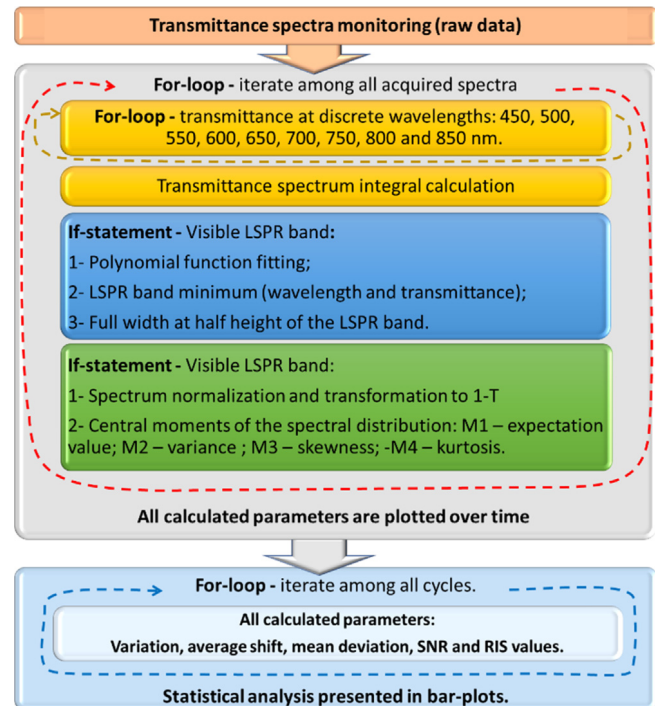


Fig. 2. Software workflow.

reliability, quick response time, low energy consumption and low environmental impact, has given rise to new materials, new detection mechanisms and new production methods [3].

With NANOPTICS, it is possible to analyse multiple parameters and quantities related to LSPR band changes occurred during sensor testing. First, the developed algorithm calculates the area under the transmittance spectra and the transmittance value (%) for discrete wavelengths. The LSPR band analysis continues by fitting the transmittance spectra with a polynomial function, to find its minimum with high resolution. The statistical analysis of the LSPR band is also provided by computing the central moments of the normalized spectral distribution of the LSPR band. All the analysed parameters are then plotted as a function of the monitoring time. Furthermore, each parameter shift (ΔP), signal-to-noise ratio (SNR) and refractive index sensitivity (RIS) are calculated using statistical analysis for several cycles.

All the examples used throughout the presented work are the result of a monitoring setup using a plasmonic thin film sensor (gold nanoparticles embedded in titanium dioxide), that was alternately exposed to vacuum (the reference condition) and an oxygen atmosphere.

2. Software description

2.1. Software architecture

NANOPTICS runs under a 64-bit Windows platform with MATLAB Runtime R2018a (v 9.4). The software is controlled by a Graphical User Interface (GUI) that allows the selection of the transmittance spectra file and the definition of initial parameters (Fig. 1). This GUI was developed using App Designer under MATLAB (R2018a 64-bit). The complete automated analysis starts after 3 main steps. In the first step ("1" in Fig. 1) the user needs to choose a ".txt" file containing the acquired transmittance spectra (raw data). The first spectrum of the monitoring is loaded and appears in GUI.

In the second step ("2" in Fig. 1), the user must set some initial parameters (from "A" to "N") for the analysis. In "A" the user starts by writing the monitoring name that will be added to the produced figures and files to export. In "B" the user must choose whether a LSPR band exists, or not. In "C", and only if the LSPR band exists and is clearly seen, the user can choose to let the algorithm select the range of the LSPR band automatically. If there is no clear observation of the LSPR band, or if the user prefers to

choose another range, the wavelength minimum and maximum can be made manually in "D" and "E". For a more reliable analysis, the lower wavelength limit should be set to include the relative maximum transmittance ("D" dash-line in the spectrum of Fig. 1). The higher wavelength limit ("E" dash-line in Fig. 1) should be set, when possible, as having the same y-axis coordinate of the lower limit. This way, only the range of interest for the LSPR band will be used for the statistical moments' analysis. In "F" the user may choose to save individual figures for all the spectra analysed. It will show all transmittance spectra acquired during monitoring, as well as the analysed range with the correspondent fitting obtained using a polynomial function. If "F" is not selected, only the first two spectra, along with the last one, will be saved as figures. By default, this checkbox is not selected since this operation is time consuming. In "G", the elapsed time between each acquired spectrum must be written, in milliseconds. In "H" the user can choose to apply a linear drift correction of the monitored parameters (details in Section 2.2). In "I" the total number of cycles (each cycle with the reference and the test conditions) must be input. Important to mention is that the last cycle of the sensing test must be done only with the reference condition, as it was programmed to be the "control" cycle. Furthermore, the time for each condition (half-cycle) should be the same, and the total cycle time should also be constant. In "J" and "K" the user must enter the portion (percentage) of each half-cycle of the monitoring to be considered for the average parameter shift calculation (details in Section 2.2). Finally, if the user is running a monitoring process to assess the refractive index sensitivity (RIS) of the sensor, the sensitivity check box ("L") needs to be ticked first, and then the refractive indices of the reference and test environments have to be input in "M" and "N" boxes.

Finally, the user presses the "Run analysis" button ("3" in Fig. 1) to start the MATLAB routines, according to the algorithm workflow diagram of Fig. 2. At the end, several text files and figures will be produced and saved in several formats.

The core algorithm (summarized in Fig. 2) takes the transmittance spectra acquired in a monitoring test (usually hundreds

of spectra) and performs several operations. Considering that all options are correctly selected, the algorithm loops through all the spectra and calculates: (i) the transmittance at several wavelengths; (ii) the integral of each spectrum; (iii) the LSPR band position (i.e. transmittance minimum, and corresponding wavelength, coordinates); (iv) the full width at half height (FWHH), and (v) the first four central moments of the normalized spectral distribution. The calculation of these parameters is explained in detail in Sections 2.2 and 2.3. All the calculated parameters are then plotted vs. time and automatically exported in several formats (details in Section 2.2). Then, considering cycles between two dielectric environments (reference, and test environments) for all the previously calculated parameters, the algorithm computes: (i) the difference in the monitored parameter values when the sensor is placed in contact with the reference and test environments, for each cycle; (ii) the grand average value and the mean deviation considering all the cycles (except the last one); (iii) the SNR values; and (iv) the RIS of the sensor (when applicable). Finally, the algorithm produces and saves figures with bar-plots considering the variation of all parameters, the SNR and the RIS values, and saves all the parameters in text files.

The GUI also makes it possible to save and load the values that were input in step “2”, by, respectively, pressing the “Save user values” or “Load user values” buttons.

2.2. Software functionalities

As mentioned in the previous section, NANOPTICS calculates several parameters, each one of them for a specific purpose. The transmittance at several discrete wavelengths is taken at specific wavelengths, 450, 500, 550, 600, 650, 700, 750, 800 and 850 nm (Fig. 3-a). This parameter is essential to assess the usability of a sensor with only a certain specific wavelength. The integral of the full spectra (Fig. 3-a) is computed using the trapezoidal numerical integration method (“trapz” function), using as limits either automatic mode or values selected by the user in the GUI (parameter “C”). Then, the contributions to this integral are divided in the left integral and the right integral to understand what happens at the left and the right sides of the LSPR band.

The position of the LSPR band minimum and the FWHH of the LSPR (transmittance) band, are estimated by fitting the LSPR band with a polynomial function (Fig. 3-b-i). The fitting function is a 9th order polynomial to ensure the best fit to the acquired data (“fit” function was used, with “fitType” parameter set to “poly9”). After the fitting, the minimum of this polynomial function is easily calculated (Fig. 3-b-ii), both the wavelength and transmittance coordinates (using the “fminbnd” function). The FWHH (Fig. 3-b-ii) is calculated considering the transmittance (y-coordinate) position at half height of the LSPR band, and then calculating the distance between the intersection of the fitted function with a horizontal line at that y position (Fig. 3-b-ii). For this parameter, the contribution at the left and right side of the LSPR band minimum is also calculated. By calculating these parameters, the physics of the sensor/analyte interaction can be analysed more deeply and with a better precision than taking the values from the raw data.

The central moments calculation of the LSPR band spectral distribution (Fig. 3-c) gives a normalized analysis for the comparison of different sensing systems and will be explained with more detail in Section 2.3.

After the calculation of all the parameters over time (Fig. 3-d), and if several cycles were done, the algorithm will perform several calculations and produce bar-plots with error bars (Fig. 3-e).

If the LSPR spectroscopy system is not used in a rigorously controlled environment, small temperature and/or pressure variations may cause a drift in the spectra measurements (Fig. 3-d-i).

To anticipate those situations, the software is prepared with a “Drift correction” option. If it is selected in the GUI interface, the calculations are done after each monitored parameter is firstly corrected (Fig. 3-d-ii) for the linear drift (“detrnd” function).

The shift of each parameter is calculated by taking the parameter vector of the test condition (second grey circle in each cycle of Fig. 3-d-ii) and subtracting the same parameter vector (of the same cycle), of the reference condition (first grey circle in each cycle of Fig. 3-d-ii). The selected range is defined by the user in the GUI as explained in the previous section. With the resulting “shift vector” of each cycle, the mean shift and standard deviation are calculated for the mentioned bar-plots and error bars (Fig. 3-e), respectively. Furthermore, the grand average shift (ΔP) and mean deviation considering all the cycles (excluding the last one) are written in the same figures for each parameter. If the “sensitivity” box is selected in the GUI, the RIS value is calculated by simply dividing the grand average shift of each parameter by the refractive index difference (Δn_i) between the reference and test materials ($RIS = \Delta P / \Delta n_i$), and added to figures (Fig. 3-e). SNR values for each parameter are calculated dividing the root mean square (RMS – “rms” function) of the “shift vector” (signal) by the RMS of the noise, which is the vector resulting from the subtraction of each value of the “shift vector” by the average of the same vector ($SNR = RMS(\text{shift vector}) / RMS(\text{shift vector} - \langle \text{shift vector} \rangle)$). To compare the full spectral range and produce an observable variable when the sensor is submitted to the reference and test environments, the optical transmittance change (OTC) is also calculated. The algorithm simply subtracts the average of the transmittance spectra in each cycle, using the same time range selected in “J” and “K” (Fig. 1), thus producing the resulting spectra changes. Examples of the produced figures can be found in the supplementary material.

The figures are saved in 3 formats: Portable Network Graphics (PNG) format for quick sharing (universal and lossless data compression), Scalable Vector Graphics (SVG) to allow high quality scaling for publication and the standard MATLAB figure format (FIG) for editing later using MATLAB. Furthermore, text files (“.txt”) with all the analysed parameters are also saved for future reference.

2.3. Sample code snippet analysis

The code snippet chosen for this implementation is presented in Listing 1 and it is about the calculation of the first four statistical central moments of the spectral distributions. Explicit equations can be found in the supplementary material.

The shape of a distribution can be described by its several statistical central moments. Before calculating the central moments, each spectrum (Fig. 3-a) is normalized (Fig. 3-c). For convenience, the spectral distribution of (1-T) is considered, where $T(\lambda)$ is the transmittance spectrum. The normalization is applied to both wavelength and “1-T”, which yields the distribution function $P(\lambda)$. Afterwards the central moments of $P(\lambda)$ are calculated (see example in Section 3). The expectation value, or first moment (M1), indicates the “centre of gravity” of a distribution. The second moment (M2), is the variance, which indicates the width or deviation. The third moment (M3), skewness, indicates any asymmetric ‘leaning’ to either left or right. Finally, kurtosis, the fourth moment (M4), indicates the degree of central “peakedness” or, equivalently, the ‘fatness’ of the outer tails (tail shape).

3. Illustrative example

A plasmonic thin film consisting of gold nanoparticles embedded in a titanium dioxide matrix (Au-TiO₂) was used in this

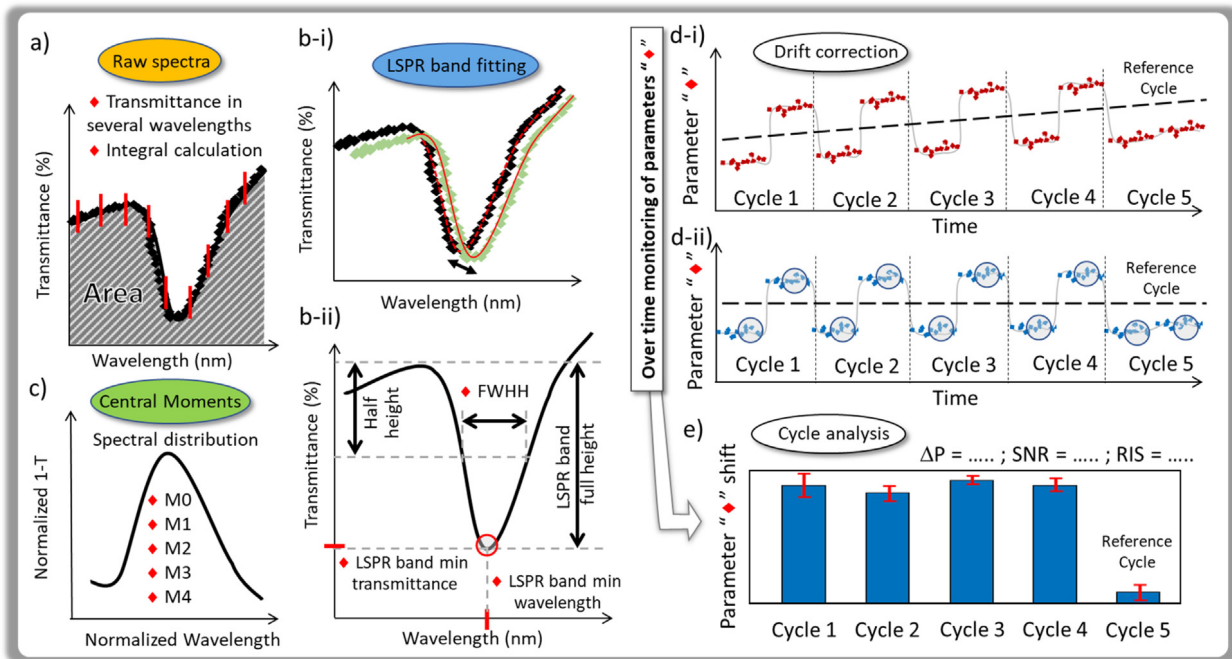


Fig. 3. Example of some parameters (marked with a red diamond) analysed by NANOPTICS: (a) transmittance coordinate at several discrete wavelengths and spectrum integral; (b) LSPR band fitting (b-i) for wavelength and transmittance coordinates at transmittance minimum (red circle) and FWHH calculation (b-ii); (c) normalized spectral distribution for central moments calculation; (d) plot of the calculated parameters during monitoring time (d-i) and after drift correction (d-ii); (e) statistical cycle analysis with parameter shift (ΔP), SNR and RIS.

```

%% Central Moments
if CentMom==1 & Peak ==1
    %Prepare data for Central Moments analysis
    wBandcutPeak1CMom=wBandcutPeak1/wBandcutPeak1(1,1); %Wavelength normalization
    TBandcutPeak1CMom=1-TBandcutPeak1/100; %from 100 % to 1 and then "1-T" normalization
    %M0 calculation - 1-T data trapezoidal method integral
    Mzero=trapz(wBandcutPeak1CMom,TBandcutPeak1CMom);
    %Transmittance data (1-T) normalization using M0
    pseudopdf=TBandcutPeak1CMom./Mzero; %defines "distribution function" "P(λ)"
    Mzero=Mzero/Mzero(1); %Mzero normalization to first integral
    %M1 calculation - Expectation value
    M1Notcentred=wBandcutPeak1CMom.*pseudopdf;
    M1=trapz(wBandcutPeak1CMom,M1Notcentred);
    %M2 calculation - Variance or deviation
    M2Notcentred=(wBandcutPeak1CMom-M1).^2.*pseudopdf;
    M2=trapz(wBandcutPeak1CMom,M2Notcentred);
    %M3 calculation - Skewness - lack of simetry. If positive left
    %shifted, if negative, right shifted
    M3Notcentred=(wBandcutPeak1CMom-M1).^3.*pseudopdf;
    M3=trapz(wBandcutPeak1CMom,M3Notcentred)./M2.^(3/2);
    %M4 calculation - Kurtosis - tail shape. If positive, heavy tailed
    %distribution, more than an exponential, if negative, the opposite
    M4Notcentred=(wBandcutPeak1CMom-M1).^4.*pseudopdf;
    M4=trapz(wBandcutPeak1CMom,M4Notcentred)./M2.^(2);
end

```

Listing 1. Central moments code snippet

work to demonstrate the functionalities of NANOPTICS (all outputs are in supplementary material). Details on the thin film sensor and the T-LSPR sensing system (hardware) can be found elsewhere [4]. To run the full analysis of 2400 spectra, presented in this work, it took 120 s on a computer with an Intel® Core™ i7 Quad Core processor (i7-8550U, 1.80 GHz, 8 MB), Intel® UHD Graphics 620, 8 GB DDR4 SDRAM and 256 GB solid state drive.

The special case of the central moments' analysis can be found in Fig. 4. The raw transmittance spectrum is shown in Fig. 4-a, and the correspondent normalized spectral distribution in Fig. 4-b. The striking result in Fig. 4-c is that all central moments change

when a test gas (oxygen) is introduced, though with different amplitudes and noise levels. Therefore, the user can choose the most sensitive sensor and the best transduction parameter not only through this normalized analysis, but also by using the SNR calculation. As an example, after the statistical analysis of all the cycles (Fig. 4-d), one can select the parameter with the best SNR value, which in this case is Kurtosis (M4), with a SNR of 18, that also coincides with the parameter with the highest RIS (absolute value of 4.2 RIU^{-1}).

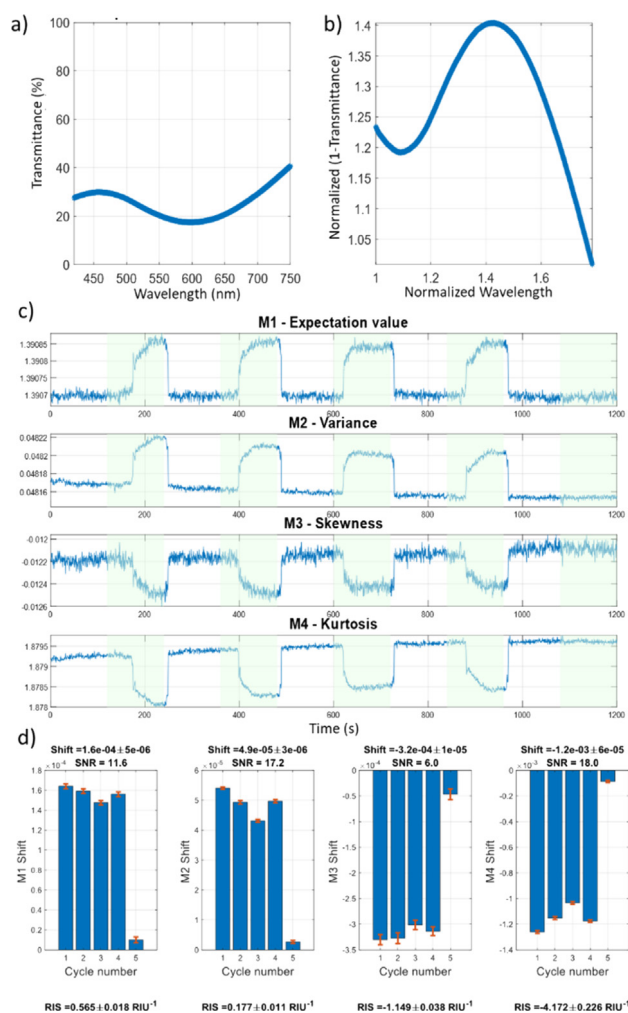


Fig. 4. (a) analysed portion of the spectra. (b) normalized spectral distribution for the calculation of the central moments. (c) central moments' over time monitoring and (d) individual cycle analysis with average shift (and corresponding mean deviation) and SNR values.

4. Impact

Molecules interact with thin film sensors by adsorption (physisorption or chemisorption), or chemical reaction. These processes lead to a physical change in a sensor film that can be detected by variations in conductivity [5,6], mass [7], work function [8] or optical properties [4,9–11]. As with all types of sensors, each thin film sensor has a certain response and recovery time, a detection limit and a saturation level that needs to be investigated during their development. The optimization of new sensing materials and systems requires fast and thorough ways of analysing their response when submitted to a stimulus.

Monitoring the LSPR band in the transmittance spectra can be done by observing different quantities in the optical absorbance and transmittance spectra. For example: (i) the wavelength of the transmittance minimum of the LSPR band; (ii) the area below the spectrum, (iii) the OTC, and (iv) the absorbance or transmittance at different wavelengths [12–18].

The presented software solves the problem of analysing the LSPR bands of plasmonic nanomaterials in monitoring processes, with hundreds of transmittance spectra, in a few seconds. It was conceived mainly for high-resolution LSPR spectroscopy systems. A statistical analysis of the LSPR bands with normalized spectral distributions is performed to compare different sensing

platforms, and both the SNR and RIS values are calculated for each analysed parameter. Furthermore, all the spectra are fitted with a polynomial function, which enables a fast and direct analysis of the transmittance (LSPR band) minimum.

This way, NANOPTICS will push forward the new generation of optical sensors, by providing an in-depth analysis tool for researchers working in new nanomaterials, particularly those exhibiting localized surface plasmon resonances (LSPR). The analysis of the LSPR band through the transmittance spectra was chosen since the measurement is straightforward.

5. Conclusions

This software enables non-MATLAB users to profit from the potentialities of data analysis and signal processing to new T-LSPR sensing platforms. The outputs of the software will accelerate further development of this type of sensors, and guide future research through the most promising pathways.

Several parameters are computed over time after a LSPR band monitoring. An emphasis was put on the statistical moments of the spectral distribution to enable a normalized comparison of the T-LSPR sensor candidates. All the calculated parameters are presented in several figures, nevertheless, if the user prefers to present the results in a different format, all the data are exported to “txt” files.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the Portuguese Foundation for Science and Technology (FCT) in the framework of the Strategic Funding UIDB/04650/2020; and by the projects NANOSSENSING POCI-01-0145-FEDER-016902, with FCT reference PTDC/FIS-NAN/1154/2014; and project NANO4BIO POCI-01-0145-FEDER-032299, with FCT reference PTDC/FIS-MAC/32299/2017. Marco S. Rodrigues acknowledges FCT for his PhD Scholarship, SFRH/BD/118684/2016. Joel Borges acknowledges FCT for his Researcher Contract from project NANO4BIO.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.softx.2020.100522>.

References

- [1] Göpel W. *Sensors Actuators* 1989;16:167–93.
- [2] Neri G. *Chemosensors* 2015;3:1–20.
- [3] Abdul S, Judit T, Ilona F, Nikolett M. *Fundam. nanoparticles*. Elsevier; 2018, p. 485–519.
- [4] Rodrigues MS, Borges J, Proença M, Pedrosa P, Martin N, Romanyuk K, Kholkin AL, Vaz F. *Nanotechnology* 2019;30:225701.
- [5] Schierbaum KD, Vaihinger S, Göpel W, van den Vlekert HH, Kloeck B, de Rooij NF. *Sensors Actuators B* 1990;1:171–5.
- [6] Xu K, Fu C, Gao Z, Wei F, Ying Y, Xu C, Fu G. *Instrum Sci Technol* 2018;46:115–45.
- [7] Pinto RMR, Brito P, Chu V, Conde JP. *J Microelectromech Syst* 2019;28:390–400.
- [8] Behzadi pour G, Fekri aval L. *Results Phys* 2017;7:1993–9.
- [9] Proença M, Rodrigues MS, Borges J, Vaz F. *Coatings* 2019;9:337.
- [10] Castillero P, Roales J, Lopes-Costa T, Sánchez-Valencia J, Barranco A, González-Elipe A, Pedrosa J. *Sensors* 2016;17:24.
- [11] Xu R, Zhao W, Xu S, Yan L, Wu Z, Liu Y, Ma W, Chen P. *J Spectrosc* 2018;2018:2985165.

- [12] Manera MG, Spadavecchia J, Buso D, de Julián Fernández C, Mattei G, Martucci A, Mulvaney P, Pérez-Juste J, Rella R, Vasanelli L, Mazzoldi P. *Sensors Actuators B* 2008;132:107–15.
- [13] Bingham JM, Anker JN, Kreno LE, Van Duyne RP. *J Am Chem Soc* 2010;132:17358–9.
- [14] Della Gaspera E, Mura A, Menin E, Guglielmi M, Martucci A. *Sensors Actuators B* 2013;187:363–70.
- [15] Gaiardo A, Fabbri B, Giberti A, Guidi V, Bellutti P, Malagù C, Valt M, Pepponi G, Gherardi S, Zonta G, Martucci A, Sturaro M, Landini N. *Sensors Actuators B* 2016;237:1085–94.
- [16] Yang Z, Sassa F, Hayashi K. *ACS Sensors* 2018;3:1174–81.
- [17] Rodrigues MS, Borges J, Vaz F. *Materials (Basel)* 2020;13:870.
- [18] Proença M, Rodrigues MS, Borges J, Vaz F. *Anal Chem* 2020;92:4349–56.