



Machine learning approach for single molecule localisation microscopy

SILVIA COLABRESE,¹ MARCO CASTELLO,² GIUSEPPE VICIDOMINI,^{2,3,4} AND ALESSIO DEL BUE^{1,3,5}

¹Visual Geometry and Modelling (VGM) Lab, Istituto Italiano di Tecnologia (IIT), Via Morego 30, Genoa, 16163, Italy

²Molecular Microscopy and Spectroscopy, Istituto Italiano di Tecnologia (IIT), Via Morego 30, Genoa, 16163, Italy

³These authors contributed equally to this work

⁴giuseppe.vicidomini@iit.it

⁵alessio.delbue@iit.it

Abstract: Single molecule localisation (SML) microscopy is a fundamental tool for biological discoveries; it provides sub-diffraction spatial resolution images by detecting and localizing "all" the fluorescent molecules labeling the structure of interest. For this reason, the effective resolution of SML microscopy strictly depends on the algorithm used to detect and localize the single molecules from the series of microscopy frames. To adapt to the different imaging conditions that can occur in a SML experiment, all current localisation algorithms request, from the microscopy users, the choice of different parameters. This choice is not always easy and their wrong selection can lead to poor performance. Here we overcome this weakness with the use of machine learning. We propose a parameter-free pipeline for SML learning based on support vector machine (SVM). This strategy requires a short supervised training that consists in selecting by the user few fluorescent molecules (~ 10-20) from the frames under analysis. The algorithm has been extensively tested on both synthetic and real acquisitions. Results are qualitatively and quantitatively consistent with the state of the art in SML microscopy and demonstrate that the introduction of machine learning can lead to a new class of algorithms competitive and conceived from the user point of view.

© 2018 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

OCIS codes: (180.2520) Fluorescence microscopy; (100.0100) Image processing; (100.5010) Pattern recognition.

References and links

1. S. W. Hell, S. J. Sahl, M. Bates, X. Zhuang, R. Heintzmann, M. J. Booth, J. Bewersdorf, G. Shtengel, H. Hess, P. Tinnefeld, A. Honigsmann, S. Jakobs, I. Testa, L. Cognet, B. Lounis, H. Ewers, S. J. Davis, C. Eggeling, D. Klenerman, K. I. Willig, G. Vicidomini, M. Castello, A. Diaspro, and T. Cordes, "The 2015 super-resolution microscopy roadmap," *J. Phys. D: Appl. Phys* **48**, 443001 (2015).
2. S. J. Sahl, S. W. Hell, and S. Jakobs, "Fluorescence nanoscopy in cell biology," *Nat. Rev. Mol. Cell Biol.* **18**, 685–701 (2017).
3. A. Small and S. Stahlheber, "Fluorophore localization algorithms for super-resolution microscopy," *Nat. Methods* **11**, 267–279 (2014).
4. H. Deschout, F. C. Zanacchi, M. Mlodzianoski, A. Diaspro, J. Bewersdorf, S. T. Hess, and K. Braeckmans, "Precisely and accurately localizing single emitters in fluorescence microscopy," *Nat. Methods* **11**, 253–266 (2014).
5. E. Betzig, G. H. Patterson, R. Sougrat, O. W. Lindwasser, S. Olenych, J. S. Bonifacino, M. W. Davidson, J. Lippincott-Schwartz, and H. F. Hess, "Imaging intracellular fluorescent proteins at nanometer resolution," *Science* **313**, 1642–1645 (2006).
6. M. J. Rust, M. Bates, and X. Zhuang, "Stochastic optical reconstruction microscopy (storm) provides sub-diffraction-limit image resolution," *Nat. Methods* **3**, 793 (2006).
7. E. Abbe, "Beiträge zur theorie des mikroskops und der mikroskopischen wahrnehmung," *Archiv für mikroskopische Anatomie* **9**, 413–418 (1873).
8. H. Shroff, C. G. Galbraith, J. A. Galbraith, and E. Betzig, "Live-cell photoactivated localization microscopy of nanoscale adhesion dynamics," *Nat. Methods* **5**, 417–423 (2008).
9. H. Nyquist, "Certain topics in telegraph transmission theory," *Transactions of the American Institute of Electrical Engineers* **47**, 617–644 (1928).

10. C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning* **20**, 273–297 (1995).
11. H. Guo, "A simple algorithm for fitting a gaussian function," *IEEE Signal Process. Mag.* **28**, 134–137 (2011).
12. S. Colabrese, M. Castello, G. Vicidomini, and A. Del Bue, "Learning-based approach to boost detection rate and localisation accuracy in single molecule localisation microscopy," in "Image Processing (ICIP), 2016 IEEE International Conference on" (IEEE, 2016), pp. 3184–3188.
13. M. Ovesný, P. Křížek, J. Borkovec, Z. Švindrych, and G. M. Hagen, "Thunderstorm: a comprehensive imagej plug-in for palm and storm data analysis and super-resolution imaging," *Bioinformatics* **30**, 2389–2390 (2014).
14. D. Sage, H. Kirshner, T. Pengo, N. Stuurman, J. Min, S. Manley, and M. Unser, "Quantitative evaluation of software packages for single-molecule localization microscopy," *Nat. Methods* **12**, 717–724 (2015).
15. S. Cox, E. Rosten, J. Monypenny, T. Jovanovic-Taliman, D. T. Burnette, J. Lippincott-Schwartz, G. E. Jones, and R. Heintzmann, "Bayesian localization microscopy reveals nanoscale podosome dynamics," *Nat. Methods* **9**, 195–200 (2012).
16. Y. S. Hu, X. Nan, P. Sengupta, J. Lippincott-Schwartz, and H. Cang, "Accelerating 3b single-molecule super-resolution microscopy with cloud computing," *Nat. Methods* **10**, 96–97 (2013).
17. Y. Tang, J. Hendriks, T. Gensch, L. Dai, and J. Li, "Automatic bayesian single molecule identification for localization microscopy," *Scientific Reports* **6**, 33521 (2016).
18. U. Köthe, F. Herrmannsdörfer, I. Kats, and F. A. Hamprecht, "SimpleSTORM: A fast, self-calibrating reconstruction algorithm for localization microscopy," *Histochem. Cell Biol.* **141**, 613–627 (2014).
19. S. Boyd and L. Vandenberghe, *Convex Optimization* (Cambridge university press, 2004).
20. R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *J. Mach. Learn. Res.* **9**, 1871–1874 (2008).
21. C. S. Smith, N. Joseph, B. Rieger, and K. A. Lidke, "Fast, single-molecule localization that achieves theoretically minimum uncertainty," *Nat. Methods* **7**, 373–375 (2010).
22. "2013 isbi grand challenge localization microscopy," <http://bigwww.epfl.ch/smlm/challenge2013/index.html>. Accessed: 12-12-2016.
23. N. Banterle, K. H. Bui, E. A. Lemke, and M. Beck, "Fourier ring correlation as a resolution criterion for super-resolution microscopy," *J. Struct. Biol.* **183**, 363–367 (2013).
24. R. Nieuwenhuizen, K. Lidke, M. Bates, D. Puig, D. GrÅijnwald, S. Stallinga, and B. Rieger, "Measuring image resolution in optical nanoscopy," *Nat. Methods* **10**, 557–562 (2013).
25. S. Colabrese, "SVM for Single Molecule Localisation Microscopy," figshare (2018) [retrieved 14 March 2018], <http://doi.org/10.6084/m9.figshare.5977453>.

1. Introduction

Single molecule localisation (SML) microscopy is an invaluable tool for biological discoveries [1, 2], however its success in obtaining a super-resolved (or nanoscopy) image depends on the optimization of several parameters, both from the experimental and image processing sides [3, 4]. With SML microscopy (e.g. PALM [5] and STORM [6]) fluorescent molecules are not active all together, they are activated at random space and time, with the aim to gather a stack of low resolution images made of few separated molecules each. A processing pipeline is then fundamental in order to obtain a super-resolved final image, circumventing the Abbe diffraction limit [7]. Generally, a SML algorithm achieved super-resolution in two main steps: a first phase of detection, or segmentation, where fluorescent molecules are localised at a pixel level, and a subsequent position refinement through fitting algorithms or faster and less accurate techniques, like centroids. These two phases are typically preceded by low-pass or band-pass filters in order to lower spurious signals contribution.

Despite such consolidated pipeline, the detection step is often made of heuristics e.g. the user is asked to select a threshold to discriminate active molecules based on image intensity, signal-to-noise ratio or the number of collected photons. The two extrema of a bad threshold selection are: an under-sampling of the biological structure, that cannot thus be recovered, or the presence of artefacts. The first effect has been experimentally demonstrated by [8] and it is in agreement with the Nyquist sampling theorem [9]; the second is due to background mistaken for active molecule, whenever the threshold is locally too low.

From a machine learning perspective, we can set the detection as a binary classification problem that can be addressed with a plethora of algorithms not yet exploited in SML microscopy. Among those we choose support vector machine (SVM) [10] because is able to obtain a spatial filter directly modelled on part of the acquired data, thus ideally being more robust to the acquisition

variability that naturally characterize these images. Moreover, to finally localise the molecules with the required accuracy, we suggest the use of a linear and recursive formulation of non-linear Least Squares fitting, similarly to what originally proposed in [11].

This work is based on the learning framework investigated in [12]. Here, the proof of concept is extended and the pipeline of [12] is embedded within a GUI in order to be usable for the community. The efficacy of the approach is proved on real datasets and performance are compared with ThunderSTORM [13], the best proposal on SML low density images according to [14]. Though ThunderSTORM requires the user to select some parameters, it has also a *default* setting for non-experts.

There are other works aiming at the automation of the parameters selection, they rely mainly on Bayesian framework (like [15], [16], [17]), or statistical significance, as in SimpleSTORM, [18]. The proposed approach is more intuitive and it is not based on any assumption on noise statistics: our hypothesis is that each frame can be divided in smaller pieces, each one having or not an active molecule inside. We claim, and experimentally verify, that these two groups, or classes, are linearly separable and the separation line can be computed directly on the first frames of each acquisition.

The paper is organized as follows: Section 2 presents the methodological tools we used and their implementation details, Section 3 shows results on synthetic and real datasets comparing them with ThunderSTORM, conclusions are drawn in Section 4.

2. Methods

2.1. Support vector machines for detection of candidate regions

Support vector machine (SVM) [10] belongs to the max-margin supervised machine learning family and has demonstrated good performance in several applications. The idea behind our SVM-based algorithm is that activations and background are linearly separable and the separation among them is the hyperplane below:

$$\mathbf{w}^T \mathbf{x} + b = 0, \quad (1)$$

where \mathbf{x} is each vectorized image patch lying on the hyperplane represented by the parameters \mathbf{w} and b . The way to find \mathbf{w} and b values is to formulate an optimization problem like in:

$$\min_{\mathbf{w}} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^M \psi(f(\mathbf{x}_i, \{\mathbf{w}, b\}), y_i), \quad (2)$$

where the first part of the objective function is a margin defined as the minimum distance among the closest examples of the two classes, the *support vectors*, and the hyperplane that separate them, while the second part is a regularization term, here ψ is the hinge loss function. The optimization is performed over a training dataset made of representatives of both classes. Each data object \mathbf{x}_i has an associated label $y_i \in \{-1, +1\}$ that is used to minimize the miss-classification error.

From Eq. (2) we can see how the optimization problem is convex thus algorithms to solve it are particularly efficient [19]. In any case, the learning is performed off-line at the beginning of the acquisition session, with a very limited computational burden. For the SVM implementation, we used *liblinear* [20], which implements the SVM with linear kernel and the value of the constant C (see Eq. (2)) is chosen with a k -fold cross-validation step, $k = 5$, before the training. As detailed in [12], the best patches dimension is 7×7 and its representation in Eq. (2), \mathbf{x}_i , is obtained with a simple vectorisation, which means that patch intensities are used as features.

By producing a spatial filter, the SVM algorithm both encloses pre-processing and detection. At inference, when a new image is given, a linear transformation is applied on each frame. Such transformation, formalised by a convolution of the image with a filter (the so-called SVM model),

has the peculiarity of being learned from the dataset that has to be analysed. Thus, the framework provides a custom filter with the property of being discriminative given the possible sensor noise and aberrations of the instrument. Specifically, the filter is a patch of 7×7 made of the hyperplane coefficients of Eq. (1), normalised to have $\|\mathbf{w}\| = 1$.

The normalisation is fundamental because the obtained model is not affected by the absolute intensity of the frames used for training and this is crucial in order to be consistent with a dropping in background intensity over time which affects real experiments.

The SVM model computed over a given dataset is thus an ad hoc spatial filter that is used in the pipeline to detect the activated molecules. Its output is a binary value for the assigned class, 1 if a patch has a molecule and -1 otherwise. We consider as candidates for a subsequent localisation refinement only patches labelled with 1.

2.2. Localisation at subpixel scale

Each emitting molecule is typically represented as a bivariate Gaussian function. This function is considered to be a good approximation of the point spread function of a microscope, thus localising an emitting molecule has the meaning of estimating the Gaussian peak. Two are the approaches generally proposed: Maximum Likelihood Estimation or Least Square fitting. We suggest [3] for an extensive comparison of the two methods. Based on [3], [14] and [4], we opted for a Least Squares fitting that does not need any camera noise model. However, the SVD-based detection approach proposed above is valid with any fitting methods. We do not detach from the literature for the subpixel estimation but, rather than applying a non-linear Least Squares fitting of a Gaussian, a parametrized Gaussian is linearised as proposed by [11].

In particular, each molecule is represented by:

$$G(x, y) = \frac{k}{2\pi\sigma_x\sigma_y} \cdot e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2} - \frac{(y-\mu_y)^2}{2\sigma_y^2}}, \quad k \in \mathfrak{R}^+ \quad (3)$$

where σ_x , σ_y , μ_x , and μ_y are the x , y variances and mean values of the Gaussian, respectively, while k is proportional to the number of photons.

Taking the logarithm of Eq. (3) is a convenient step because the solution of the Least Squares can be obtained from the inversion of a linear system in $\{A, B, C, D, E\}$, defined as in Eq. (4) and (5):

$$\hat{G}(x, y) = \ln \left\{ \frac{k}{2\pi\sigma_x\sigma_y} \right\} - \frac{(x - \mu_x)^2}{2\sigma_x^2} - \frac{(y - \mu_y)^2}{2\sigma_y^2} \quad (4)$$

$$= A + Bx + Cy + Dx^2 + Ey^2,$$

$$\begin{cases} A \doteq \ln \left\{ \frac{k}{2\pi\sigma_x\sigma_y} \right\} - \frac{\mu_x^2}{2\sigma_x^2} - \frac{\mu_y^2}{2\sigma_y^2} \\ B \doteq \frac{\mu_x}{\sigma_x^2} \\ C \doteq \frac{\mu_y}{\sigma_y^2} \\ D \doteq -\frac{1}{2\sigma_x^2} \\ E \doteq -\frac{1}{2\sigma_y^2} \end{cases} \quad (5)$$

The quadratic function under consideration represents a bi-dimensional Gaussian if and only if the following constraints are met:

$$\begin{cases} \forall A \in \mathfrak{R}, \\ B >= 0 \\ C >= 0 \\ D < 0 \\ E < 0 \end{cases} \quad (6)$$

The implemented algorithm verifies that these conditions hold, if not the patches are discarded. Each single patch is composed by the tuples (x_i, y_i, I_i) with x_i, y_i representing the position and I_i the grey scale intensity of the i pixel. All the elements in the tuple are positive integers, I_i because a negative value has no physical meaning, x_i and y_i because they are indexes. For a patch with N pixels, our goal is to minimize:

$$\sum_{i=1}^N \epsilon_i^2 = \sum_{i=1}^N [\hat{I}_i - \hat{G}(x_i, y_i)]^2, \quad (7)$$

where \hat{I}_i is the logarithm of I_i , so that the sum of squared differences between the model and the data is minimum.

Computing the derivative of the residual sum of squares with respect to each parameter and setting it to zero, we obtain the linear system:

$$M \cdot \begin{pmatrix} A \\ B \\ C \\ D \\ E \end{pmatrix} = \begin{pmatrix} \sum \ln z_i \\ \sum x_i \ln z_i \\ \sum y_i \ln z_i \\ \sum x_i^2 \ln z_i \\ \sum y_i^2 \ln z_i \end{pmatrix}, \quad (8)$$

where

$$M = \begin{pmatrix} N-1 & \sum x_i & \sum y_i & \sum x_i^2 & \sum y_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i y_i & \sum x_i^3 & \sum x_i y_i^2 \\ \sum y_i & \sum x_i y_i & \sum y_i^2 & \sum x_i^2 y_i & \sum y_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^2 y_i & \sum x_i^4 & \sum x_i^2 y_i^2 \\ \sum y_i^2 & \sum x_i y_i^2 & \sum y_i^3 & \sum x_i^2 y_i^2 & \sum y_i^4 \end{pmatrix}. \quad (9)$$

A good strategy is to weigh the data, since in logarithmic scale a larger error is associated with small values of I_i . Our goal (at each iteration) is now to minimize the following expression:

$$\begin{aligned} \sum_{i=1}^N w_i^2 \epsilon_i^2 &= \sum_{i=1}^N w_i^2 [\hat{I}_i - \hat{G}_k(x_i, y_i)]^2 \\ &= \sum_{i=1}^N |G_{k-1}(x_i, y_i)|^4 [\hat{I}_i - \hat{G}_k(x_i, y_i)]^2, \end{aligned} \quad (10)$$

if $k > 1$

where the weight w_i is the squared predicted value of the model obtained in the previous iteration. This leads to the following linear system:

$$\hat{M} \cdot \begin{pmatrix} A \\ B \\ C \\ D \\ E \end{pmatrix} = \begin{pmatrix} \sum w_i \ln I_i \\ \sum w_i x_i \ln I_i \\ \sum w_i y_i \ln I_i \\ \sum w_i x_i^2 \ln I_i \\ \sum w_i y_i^2 \ln I_i \end{pmatrix}, \quad (11)$$

where:

$$\hat{M} = \begin{pmatrix} \sum w_i & \sum w_i x_i & \sum w_i y_i & \sum w_i x_i^2 & \sum w_i y_i^2 \\ \sum w_i x_i & \sum w_i x_i^2 & \sum w_i x_i y_i & \sum w_i x_i^3 & \sum w_i x_i y_i^2 \\ \sum w_i y_i & \sum w_i x_i y_i & \sum w_i y_i^2 & \sum w_i x_i^2 y_i & \sum w_i y_i^3 \\ \sum w_i x_i^2 & \sum w_i x_i^3 & \sum w_i x_i^2 y_i & \sum w_i x_i^4 & \sum w_i x_i^2 y_i^2 \\ \sum w_i y_i^2 & \sum w_i x_i y_i^2 & \sum w_i y_i^3 & \sum w_i x_i^2 y_i^2 & \sum w_i y_i^4 \end{pmatrix}, \quad (12)$$

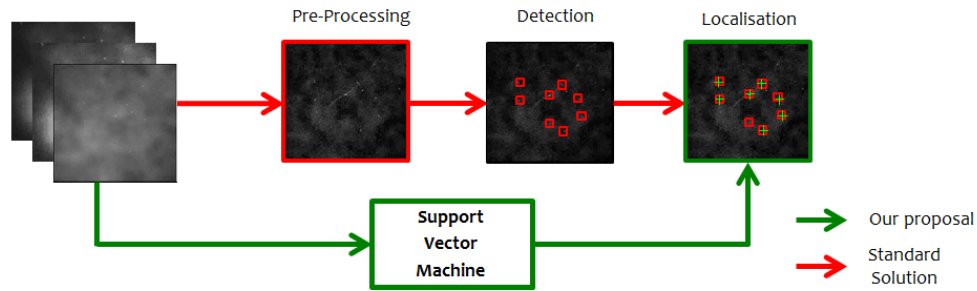


Fig. 1. Pipeline for SML. The standard solution is presented on top (red arrows) while our proposal is at the bottom, underlined with green arrows. The *Support Vector Machine* block aims at substituting the red blocks of *Pre-Processing* and *Detection*, localisation is a common step instead.

and

$$w_i = \begin{cases} |G_{k-1}(x_i, y_i)|^2 & \text{if } k > 1 \\ |G_0(x_i, y_i)|^2 & \text{if } k = 1, \end{cases} \quad (13)$$

with k being the current iteration number; $G_0(x_i, y_i)$ is an initial guess, a Gaussian centered in the origin of the detected patch under analysis. The outcome of this step is a list of points for which the iterative procedure has converged. The number of iterations that states the convergence is $N = 20$, regardless the dataset. A check on the convergence allows the algorithm to discard points for which this condition is not reached.

2.3. 2D localisation pipeline and GUI

Our proposal is sketched in Fig. 1 where it is compared with the schematic of the most common pipelines in the literature. The first green block, called *Support Vector Machine* aims at substituting the red blocks of *Pre-Processing* and *Detection*. Localisation, instead, is a common operation even if the suggested approach has some novelties with respect to what is proposed in the literature of SML microscopy [4] and [21].

The pipeline is embedded within a GUI that, together with the main functions like loading datasets or exporting results, guides the user in the training of the algorithm. The user is asked to select some active molecules while examples from the background are selected automatically. Few selections are enough to have good performance, as detailed in Section 3.

2.4. Datasets

In order to evaluate the performance of our proposal, both synthetic and real datasets are used. While synthetic datasets ease a quantitative evaluation of the algorithm, real datasets are fundamental to check the behaviour in case of real acquisition scenario.

Synthetic datasets are taken from the 2013 SML microscopy challenge described in [22]. They have various levels of difficulty, depending on the amount and kind of simulated noise while the biological structure to be imaged is mainly a bundle of Tubulin protein. Each dataset provides a collection of frames in uncompressed 16-bits TIFF format; Table 1 shows more details about them with *Tubulin II* being the dataset with the highest auto-fluorescent noise.

The proposed pipeline is then tested also on real datasets, not just to be sure that the algorithm is able to face real acquisitions, but also to test if the SVM, though a supervised algorithm, is applicable to real conditions in which the training dataset has to be created without ground truth.

The 2013 Challenge website provides access also to four experimental datasets, courtesy of the Laboratory of Experimental Biophysics in EPFL, Switzerland. We have used the acquisitions

Table 1. Properties of the synthetic datasets in use, for detailed informations on the acquisition settings refer to [22].

DATASET	# of molecules	# of frames	Resolution (nm/pixel)	Dimension
Bundled	81049	12000	100	64 × 64
TubulinI	100000	2401	150	256 × 256
TubulinII	100000	2401	150	256 × 256

Table 2. Properties of the real datasets in use, for other information (frame rate, laser wavelength, etc.) refer to [22].

DATASET	NA	# of frames	Resolution (nm/pixel)	Dimension
TubulinsLS	1.3	15000	100	64 × 64
TubulinAF647	1.46	9990	100	128 × 128

named TubulinsLS and TubulinAF647. The first one is taken imaging Tubulin filaments, the other one is a fixed cell, stained with mouse anti-alpha-tubulin primary antibody and Alexa647 secondary antibody, Tab. 2 summarizes the images acquisition conditions.

2.5. Evaluation metrics

In the SML evaluation protocol, having a true positive (TP) means that a localisation is matched with a point of the Ground Truth (GT) and this implies that it must be spatially close enough to it. Following [14], we select a radius equal to the full-width half-maximum (FWHM) of the PSF, in our case around 250 nm. The remaining localised molecules farthest from the radius, unpaired, are categorized as false positives (FP). Finally, GT molecules that are not associated are called false negatives (FN).

Detection rate and localisation accuracy are the performance indexes commonly used in SML community, together with Fourier Ring Correlation [23, 24] which is useful in the case of real datasets when ground truth is not available. The Jaccard index (JAC) is used for monitoring the detection, while the root mean square error (RMSE) is used for the accuracy. Only the detections matched with the GT contribute to the equation 14 as:

$$a^2 = \frac{1}{N} \sum_{n=1}^N (x_n^{Test} - x_n^{Ref})^2 + (y_n^{Test} - y_n^{Ref})^2, \quad (14)$$

where $N = TP$ and $a = \text{accuracy}$.

For the sake of completeness we present the results also in terms of Precision and Recall indexes, with formulas as follow: Jaccard $J = TP/(FN + FP + TP)$, Precision $p = TP/(TP + FP)$ and Recall $r = TP/(TP + FN)$.

3. Results

3.1. Synthetic data analysis

Preliminary results of the proposed method was presented in [12]. The evaluation was purely quantitative and it was performed on the synthetic datasets of 2013 SML microscopy Challenge [22]. The obtained results were compared with the best performing software of the Challenge, namely ThunderSTORM [13], revealing a good detection ability of our pipeline. Results was close or at the level of ThunderSTORM and the proposal stands out with better performance in two of the three datasets.

However, in order to obtain the spatial filter (SVM model) for the detection task, the classifier needs to be trained. To this end, in [12] each dataset was divided exactly in two non-overlapping

stacks of frames; from the first, samples of positive and negative classes are extracted based on GT information while the other frames were used as test set, to evaluate the performance of the pipeline and reconstruct the super-resolved image.

Such way of building the training set is typical in classification problems but it is not realistic for SML microscopy acquisition, questioning the usability of the approach. Here we demonstrate that there is no need for a big amount of data by explicitly characterize the behavior of the pipeline as the cardinality of the training set changes. We embed the algorithm proposed in [12] in a wider framework made of a GUI and a simpler yet effective way to create a training set. Besides being easier to use, this system allows a quick selection of few positive samples and the generation of the training dataset from them.

A drawback in not having the GT information is the need to assume the highest local maxima within a patch as the pixel location of an activation, which is not always the case due to the presence of shot noise. This workaround has an impact on the detection ability if we compare the results shown in [12] with the column named SVM-half of Table 3. In particular, such a training set composition lead to a more selective identification of activations, as the Jaccard index reflects.

Table 3. Performance comparison among ThunderSTORM (TS), our SVM-based approach with half of the acquired stack used for the training (SVM-half) and the same SVM approach using for training only 20 activations (SVM-few).

Dataset	Jacc (%)			Acc (nm)			Prec (%)			Rec (%)		
	TS	SVM half	SVM few	TS	SVM half	SVM few	TS	SVM half	SVM few	TS	SVM half	SVM few
Bundled	93	81.7	87.2	14.5	14	14.3	99.8	100	99.9	93.2	81.7	87.2
TubulinI	74.4	47.3	64.5	22.2	17.2	18	96.2	100	99.9	76.7	47.3	64.5
TubulinII	67.3	42.3	48.8	27.7	22.2	23.8	98.4	100	79.9	68.1	42.3	55.6

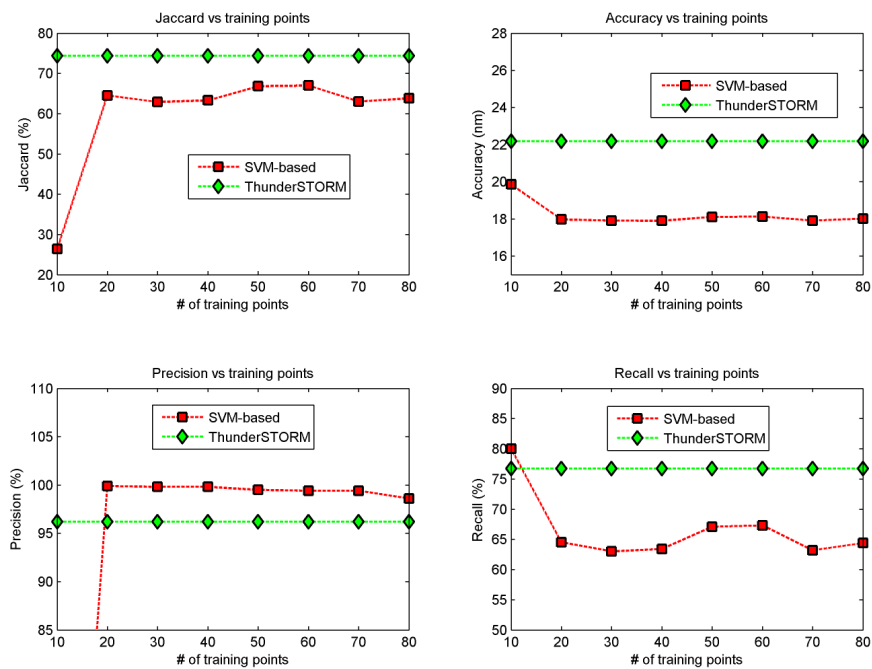


Fig. 2. Quantitative results over the synthetic dataset Tubulin I. The measurements taken into account are, clock wise starting from top-left: Jaccard, Accuracy, Precision and Recall.

Table 3 shows the performance of ThunderSTORM and the SVM-based approach with two different sizes of the training set. Namely, SVM-half means that half of the images are used to build the SVM while SVM-few referred to a manual selection, through the GUI, of 20 activations. Comparing the two kinds of training, we can say that all the performance that we can obtain with few activations are in line with what we have using half of the dataset, looking at indexes like Accuracy and Precision, and are much better in terms of Jaccard and Recall. A possible explanation is that with too much data the algorithm tends to be driven by the activation patches that are more common. Instead, the reason of the drop in Precision we have on TubulinII dataset is that the dataset has more variability, being the noisiest, and the use of only 20 points penalizes the performance.

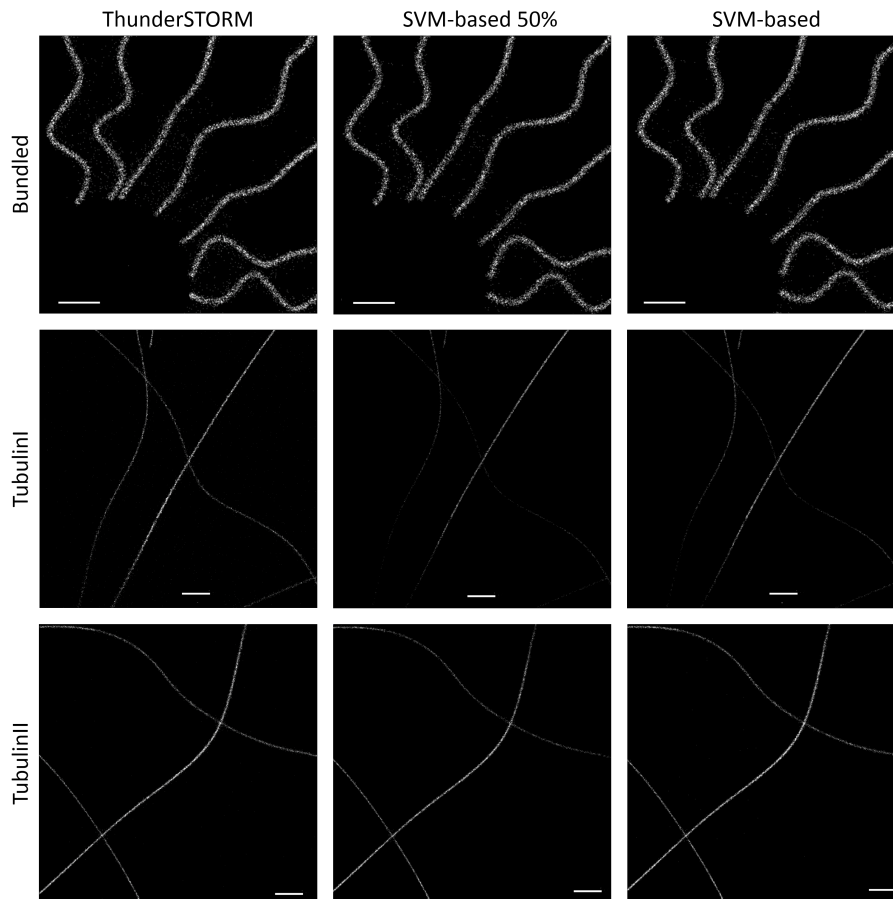


Fig. 3. Results over the synthetic datasets Bundled Of Tubulin (top), Tubulin I (centre) and Tubulin II (bottom). Our software is run neglecting the ground truth. Column wise, detail of super-resolved images obtained by ThunderSTORM, by the SVM with half of the available images for the training and by the SVM with 20 selections. The scale bar values are $0.3 \mu\text{m}$ for Bundled Of Tubulin (top), and $1.2 \mu\text{m}$ for Tubulin I and Tubulin II.

We remark that the manual selection does not need to be precise because the algorithm automatically shift the selection on the closest pixel with a local maxima intensity. Looking at this selection from a different perspective, the user is only asked to select few *areas* within one or more frames where to find positive activations, its selection will be refined by the algorithm prior to be used. For each new selected pixel, also the number of background samples increases,

so that the training set is equally divided into activations and background (i.e. it is balanced). Whenever the manual selection is completed, the SVM is trained and the processing can start on the remaining frames. Regarding the number of activations needed, Fig. 2 represents the performance of the whole pipeline when the SVM of the detection step is trained increasing the number of positive samples from 10 to 80, with a step of 10. We can see how on TubulinI with the selection of few points, the Jaccard measurement is not as favourable as in ThunderSTORM and the reason is a poor recall, while there is an improvement of Precision and Accuracy with respect to the competitor; these results are in agreement with what we observe in all the analysed datasets.

We can conclude that there is no need for the user to select a large amount of activations: decreasing their number, the performance remains stable until a value between 10-20 selections, for which there is a drop. The user can select a reasonable yet arbitrary amount of activations without impairing the processing outcome. In this context, it is important to highlight that for a not-expert user the selection of few activations is more intuitive than to chose arbitrary thresholds.

Fig. 3 shows the super-resolved images obtained by our algorithm, second and third column, and the ones reconstructed by ThunderSTORM. Images underline the strength and limit of our pipeline: the reconstruction is less noisy with respect to ThunderSTORM outputs, however not all the TPs are detected, resulting in a structure made of fewer molecules in some regions. This prudential approach is quantitative measurable and it clearly emerges observing the behaviour of Precision and Recall measurements, see Fig. 2 as example on Tubulin I dataset. However, comparison of the proposed method and ThunderSTORM via the FRC analysis does not show significant differences, as it is possible to see from Fig. 4.

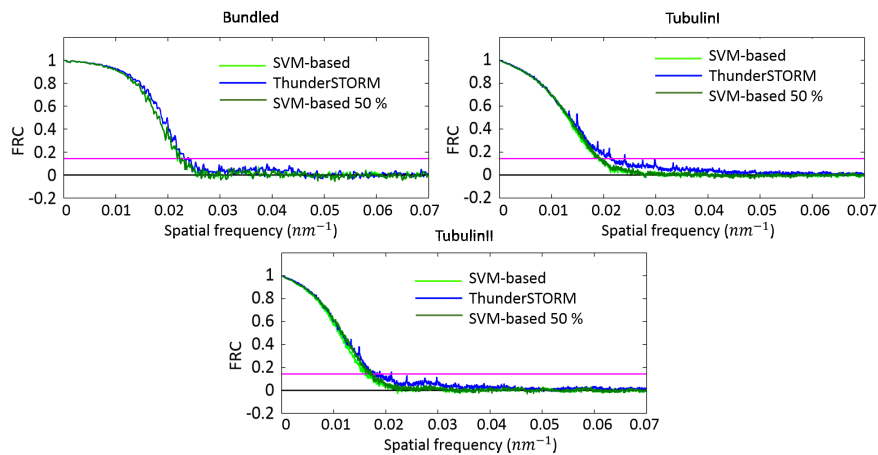


Fig. 4. Quantitative comparison of our algorithm with ThunderSTORM. The FRC is computed for the synthetic datasets, clock-wise starting from top-left: Bundled Of Tubulin, Tubulin I and Tubulin II.

3.2. Real data analysis

In the previous Section we have seen how it is possible to achieve performance in line with a state of the art algorithm, such as ThunderSTORM, without the need of a time-consuming training and without the definition of arbitrary thresholds – now replaced by an intuitive labelling procedure for the user.

We are proposing a pipeline without mentioning any hypothesis on noise or background statistical behaviour. In fact, the detection task is based on the only assumption that the two classes, activation and background, are linearly separable. In order to verify the benefits of the

approach with real acquisitions, thus real background, we run the algorithm on several Tubulin datasets.

Figure 5 presents the qualitative evaluation of our approach on TubulinAF647 and TubulinsLS datasets [22], comparing it with ThunderSTORM. While on ThunderSTORM we use the *default* set-up, on our pipeline we manually select 20 activations within the first frames. As it is evident from the FRC, Fig. 6, the performance are exactly the same on TubulinAF647 while there is a loss of few *nm* on TubulinsLS. However, the final super-resolved images present sensibly less artefacts with respect to ThunderSTORM reconstructions. This observation is valid on both datasets as shown in Fig. 5, where a magnification underlines how ThunderSTORM detected spurious activations and some big spots that our algorithm is more robust to discard. This later difference, explain also the drop in correlation at low spatial frequencies in TubulinAF647 in the FRC curve for our method with respect to ThunderSTORM.

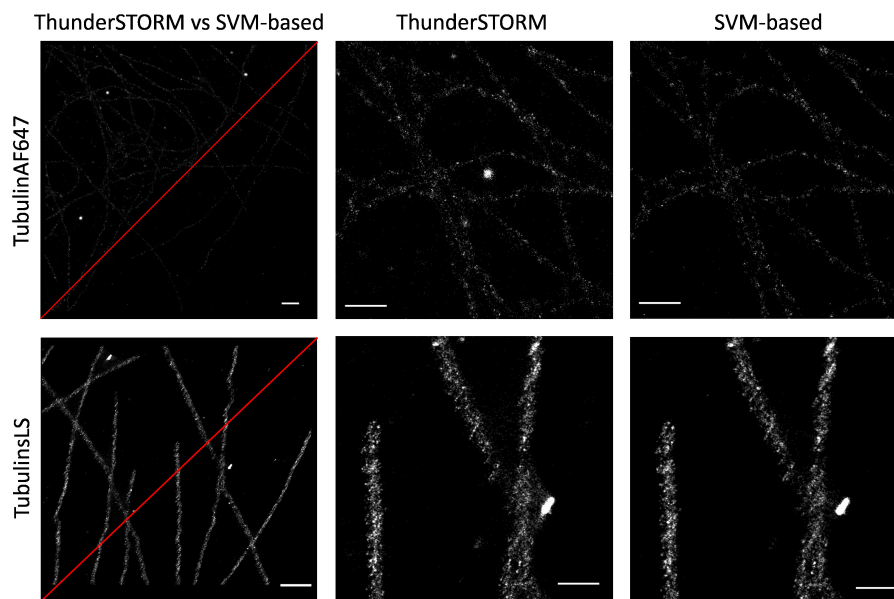


Fig. 5. Results over the real datasets TubulinAF647 (top) and TubulinsLS (bottom). Column wise, from left to right: super-resolved images obtained by ThunderSTORM, upper left corner, and our SVM-based algorithm, scale bar $1 \mu\text{m}$ (top) and $0.7 \mu\text{m}$ (bottom); a detail of ThunderSTORM reconstruction and a detail of our SVM-based algorithm, scale bar $0.6 \mu\text{m}$ (top) and $0.3 \mu\text{m}$ (bottom).

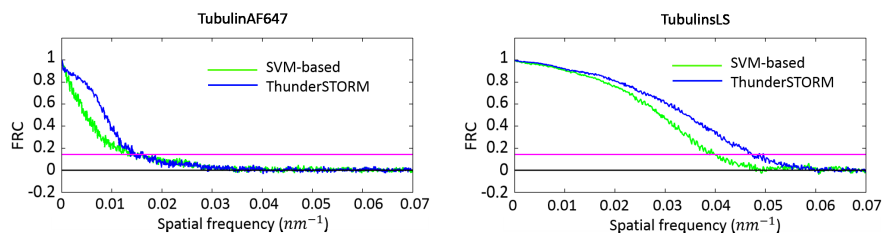


Fig. 6. Quantitative results (FRC) over the real datasets TubulinAF647 and TubulinsLS.

4. Conclusions

Within SML microscopy, processing after image acquisition is fundamental to obtain a super-resolved image. Unfortunately, state of the art algorithms rely on thresholds heuristically chosen by the user to accomplish molecules detection.

We experimentally demonstrate that it is possible to formulate the detection step as a classification problem, resulting in the building of a custom filter for each dataset to analyse. Such training step is not always necessary, filters are saved and can be reused but we advise the retraining in case the acquisition conditions differs too much. In particular, the same filters can be used to analyses a long series of experiments where microscope and sample conditions does not change, which is a typical scenario in SML microscopy.

The SVM filter has performances in line with the state of the art but it is conceived to be threshold free, learning directly from the dataset the division between molecules and background. The supply of few molecules to the algorithm prevents also the use of a pre-processing step, reducing the introduction of artefacts and the computational load of the algorithm; this is further reduced adopting the linearisation of a LS fitting, which eventually is a very accurate tool.

As future work, we plan to test our algorithm on a broader range of samples (with different signal-to-noise/background ratios) and to extend our method to 3D by accounting for PSF with distortions related to their position in depth. The detector has to be extended to encode such distortions, for instance by designing appropriate kernels. Moreover, our fitting algorithm can be possibly extended without strong modifications the algorithmic structure, especially if the PSF model is still an exponential function. Another significant challenge in the context of SML microscopy, which is not currently addressed by our SVM approach, is the ability to work under high-density molecular conditions, i.e., low sparsity. Solution to this problem can still be searched in the field of machine learning, but within other approaches, such as neural networks. The complete MATLAB source code for our SVM-based localisation approach is available from [Code 1](#) [25].

Acknowledgements

The authors are grateful to Dr. Francesca Cella Znacchi and Dr. Francesca Pennacchietti (Istituto Italiano di Tecnologia) for fruitful discussions.

Disclosures

The authors declare that there are no conflicts of interest related to this article.