Chapter Two

Quantitative Comparison of Spot Detection Methods in Fluorescence Microscopy

Not everything that can be counted counts, and not everything that counts can be counted.

— Albert Einstein (1879-1955)

Abstract — Quantitative analysis of biological image data generally involves the detection of many subresolution spots. Especially in live cell imaging, for which fluorescence microscopy is often used, the signal-to-noise ratio (SNR) can be extremely low, making automated spot detection a very challenging task. In the past, many methods have been proposed to perform this task, but a thorough quantitative evaluation and comparison of these methods is lacking in the literature. In this chapter, we evaluate the performance of the most frequently used detection methods for this purpose. These include six unsupervised and two supervised methods. We perform experiments on synthetic images of three different types, for which the ground truth was available, as well as on real image data sets acquired for two different biological studies, for which we obtained expert manual annotations to compare with. The results from both types of experiments suggest that for very low SNRs ($\approx$2), the supervised (machine learning) methods perform best overall. Of the unsupervised methods, the detector based on the so-called h-dome transform from mathematical morphology performs comparably, and has the advantage that it does not require a cumbersome learning stage. At high SNRs ($>$5), the difference in performance of all considered detectors becomes negligible.

Based upon: I. Smal, M. Loog, W. J. Niessen, E. Meijering, "Quantitative Comparison of Spot Detection Methods in Fluorescence Microscopy", *submitted.*

2.1 Introduction

The very first stage in the analysis of biological image data generally deals with the detection of objects of interest. In fluorescence microscopy, which is one of the most basic tools used in biology for the visualization of subcellular components and their dynamics [88, 100, 113, 156, 164, 180], the objects are labeled with fluorescent proteins and appear in the images as bright spots, each occupying only a few pixels (see Fig. 2.1 for sample images). Digital image analysis provides numerical data to quantify and substantiate biological processes observed by fluorescence microscopy [3, 45, 97, 185, 191]. Such automated analysis is especially valuable for high-throughput imaging in proteomics, functional genomics and drug screening [42, 103]. Nevertheless, obtaining accurate and complete measurements from the image data is still a great challenge [38]. In many cases, the quality of the image data is rather low, due to limitations in the image acquisition process. This is especially true in live cell imaging, where illumination intensities are reduced to a minimum to prevent photobleaching and photodamage, resulting in a very low signal-to-noise ratio (SNR) [53, 95, 96]. In addition, despite recent advances in improving optical microscopy [51,63], the resolution of even the best microscopes available today is still rather coarse (on the order of 100 nm) compared to the size of subcellular structures (typically only several nanometers in diameter), resulting in diffraction-limited appearance. As a consequence, it is often difficult, even for expert biologists, to distinguish objects from irrelevant background structures or noise.

In practice, automated object detection methods applied to fluorescence microscopy images either report too many false positives, thus corrupting the analysis with the presence of nonexistent objects, or they detect less objects than are in fact present, causing subsequent analyses to be biased towards more clearly distinguishable objects. This is also a serious issue in time-lapse imaging, where the objects of interest are to be tracked over time to study their dynamics. In common tracking algorithms, which consist of separate detection (spatial) and linking (temporal) stages [95, 96], the performance of the detector is crucial: poor detection likely causes the linking procedure to yield nonsensical tracks, where correctly detected objects in one frame are connected with false detections in the next (and vice versa), or where tracks are terminated prematurely because no corresponding objects were detected in the next frame(s). Modern tracking approaches, based on Bayesian estimation [141,142], avoid the hard decision thresholds in the detection stage of conventional approaches, and describe object existence in terms of probability distribution functions (pdf). Such real-valued pdfs reflect the degree of believe in the presence of an object at any position in the image in a more "continuous" fashion, in contrast with the binary representation (either "present" or "not present") obtained after applying hard thresholds. Nevertheless, even in probabilistic tracking frameworks, some form of "deterministic" object detection is still necessary in the track initiation and termination procedures [141,142,146], again illustrating the relevance of having a good spot detector. Several detectors have been proposed in the literature, and the classic, relatively simpler methods have been compared previously for tracking [26,32], but a thorough quantitative comparison including recent, more complex methods is missing.

In this study, we compare several detectors that are frequently used for object detection in fluorescence microscopy imaging, and quantify their performance using both synthetic images and real image data from different biological studies. The sensitivity of the methods is studied as a function of their parameters and image quality (expressed in terms of SNR). The methods under consideration range from relatively simple local background subtraction [185], to linear or morphological image filtering [20, 21, 128, 142, 146, 161], to wavelet-based multiscale products [52, 108], and machine learning methods [73]. They can be divided into two groups: unsupervised and supervised. The first consists of algorithms that (implicitly or explicitly) assume some object appearance model and contain parameters that need to be adjusted either manually or semi-automatically in order to get the best performance for a specific application. Supervised methods, on the other hand, "learn" the object appearance from annotated training data—usually a large number of small image patches containing only the object intensity profiles (positive samples) or irrelevant background structures (negative samples).

This chapter is organized as follows. First, in Section 2.2, we provide background information on the image formation process in fluorescence microscopy and describe the object detection framework in general. This helps to put the different detection methods in proper perspective and provides motivations for some of the choices made later on in the chapter. The detection methods that were considered in this study and that implement the general framework are described in Section 2.3. Next, in Section 2.4, we present the experimental results of applying the detection methods to synthetic images, for which ground truth was available, as well as to real fluorescence microscopy image data from several biological studies. A concluding discussion of the main findings and their implications is given in Section 2.5.

2.2 Detection Framework for Fluorescence Microscopy

2.2.1 Image formation

In fluorescence microscopy, specimens are labeled with fluorophores. The distribution of fluorescence caused by exciting illumination is then observed and captured by a photosensitive detector (usually a CCD camera or a photomultiplier tube) that measures the intensity of the emitted light and creates a digital image of the sample. The objects of interest in our application appear in images as blurred spots, which are relatively small and compact, have no clear borders (which is why we prefer to speak of "detection" rather than "segmentation"), and their intensity is higher than the background. The blurring is caused by the diffraction phenomenon and imperfections of the optical system, which for commonly used confocal microscopes limits the resolution to about 200 nm laterally and 600 nm axially [95, 161, 185, 190]. This is characterized by the point spread function (PSF) of the system, which is the image of a point source of light. In our applications, the theoretical PSF, which can be expressed by the scalar Debye diffraction integral [190], can in practice be approximated by a 2D or 3D Gaussian PSF [161], depending on the dimensionality of the image data.

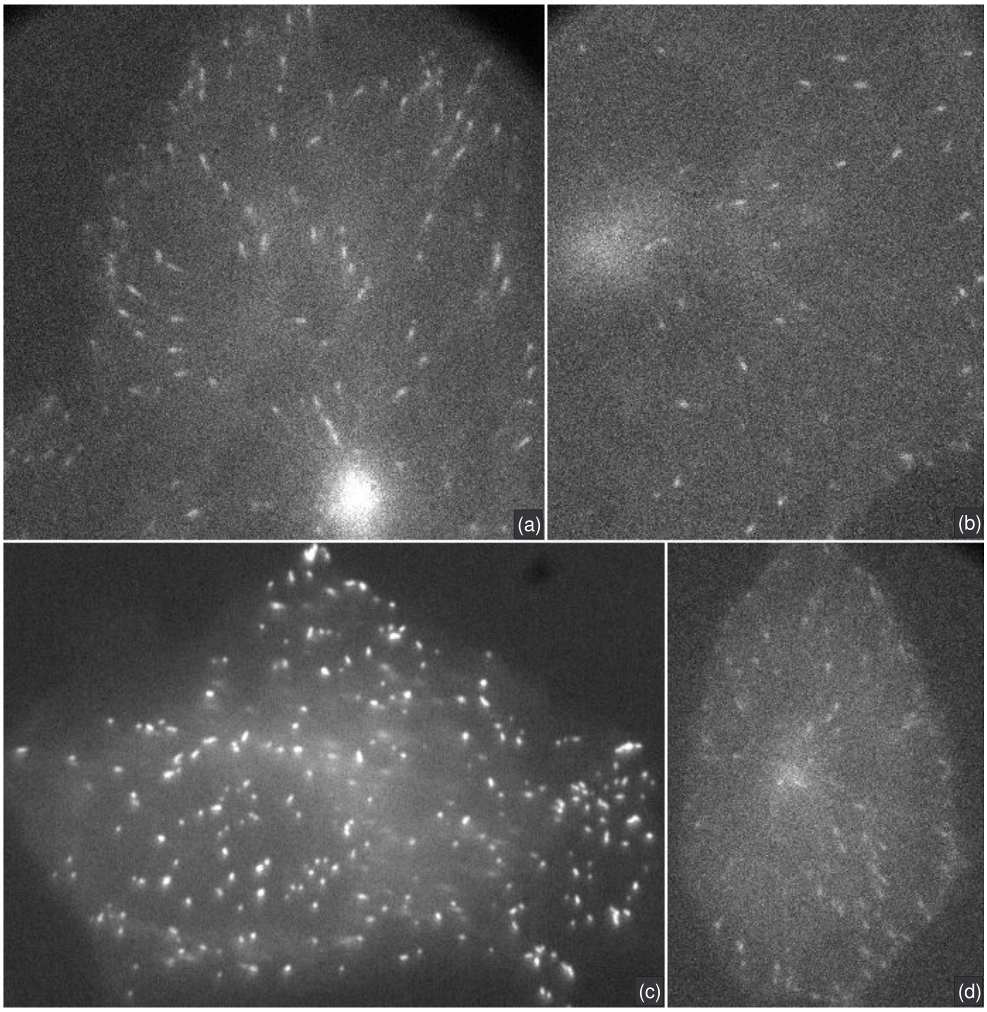

Figure 2.1. Sample images of microtubules (a,b,d) and peroxisomes (c) labeled with green fluorescent protein (GFP) and imaged using confocal microscopy. The images are single frames from 2D time-lapse studies, acquired under different experimental conditions. The quality of the images ranges from SNR $\approx$ 4–6 (a,c) to $\approx$ 2–4 (b,d).

Apart from the diffraction-limited spatial resolution, another major source of aberrations introduced in the imaging process is intrinsic photon noise, which results from the random nature of photon emission. Photon noise, which is independent of the detector electronics, can be reduced (and, consequently, the SNR increased) only by increasing the light intensity or the exposure time. However, increasing the light intensity in order to improve the image quality causes the fluorescent signal to fade permanently due to photon-induced chemical damage and covalent modification,

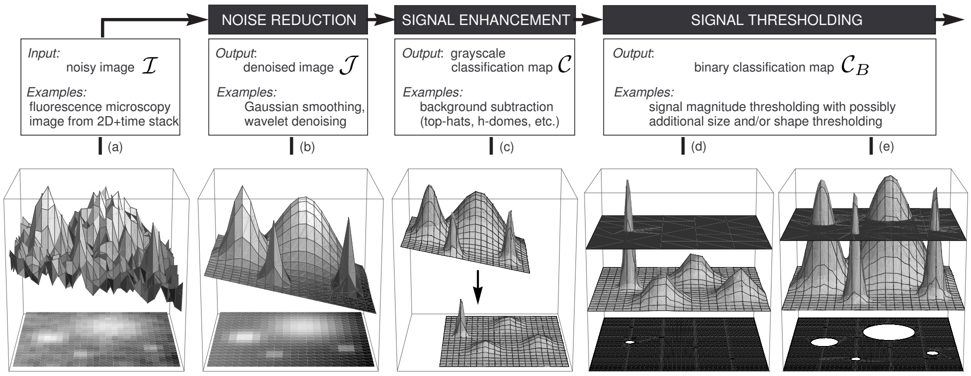
NOISE REDUCTION
SIGNAL ENHANCEMENT
SIGNAL THRESHOLDING
Input: noisy image $\mathcal{I}$
Examples: fluorescence microscopy image from 2D+time stack
Output: denoised image $\mathcal{J}$
Examples: Gaussian smoothing, wavelet denoising
Output: grayscale classification map $\mathcal{C}$
Examples: background subtraction (top-hats, h-domes, etc.)
Output: binary classification map $\mathcal{C}_B$
Examples: signal magnitude thresholding with possibly additional size and/or shape thresholding
(a)
(b)
(c)
(d)
(e)

Figure 2.2. Object detection framework. The original noisy image (a) is preprocessed with some noise reduction method, and the resulting image (b) is transformed (enhanced) into a new image (c), in which the possible object locations have higher signal magnitude than all other structures (d), or all the suspicious locations are marked (e). The threshold (represented by the dark-gray planes in (d) and (e)) is applied and the connected components in the binarized image (white clusters on the black background) are counted as the detected objects.

a process called photobleaching [185]. While this effect can be exploited to study specific dynamical properties of particle distributions [87, 156], it hampers detection and tracking of individual fluorescent particles. With a laser as excitation source, photobleaching is observed on the time scale of microseconds to seconds, and should be taken care of especially in time-lapse microscopy.

In this study, we deal with subresolution objects (blurred spots) on a possibly nonuniform background, the appearance of which can be modeled using a Gaussian approximation of the PSF. While for experimental and illustration purposes we limit ourselves to 2D image data, all detection methods considered in this chapter can be applied straightforwardly to 3D data without any substantial changes. Each image $\mathcal{I}$ consist of $N_x \times N_y$ pixels, where each pixel corresponds to a rectangular area of dimension $\Delta_x \times \Delta_y \text{nm}^2$ and the measured intensity at position (i, j) is denoted as $I(i, j)$. In other words $\mathcal{I} = \{I(i, j) : i = 1, \ldots, N_x, j = 1, \ldots, N_y\}$. In order to model different types of subcellular particles (round or elongated appearance), we use an asymmetric 2D Gaussian function. In this case, the measured intensity at (i, j) caused by the fluorescent light source located at (x, y), which is the real-valued position within the image, is given by

$$I(i,j) = B(i,j) + \exp\left(-\frac{1}{2}\mathbf{m}^T\mathbf{R}^T\mathbf{\Sigma}^{-1}\mathbf{R}\mathbf{m}\right), \tag{2.1}$$

where $\mathbf{\Sigma} = \text{diag}[\sigma^2_{\max}, \sigma^2_{\min}]$, $\mathbf{R} = \mathbf{R}(\phi)$ is a rotation matrix

$$\mathbf{R}(\phi) = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}, \quad \mathbf{m} = \begin{pmatrix} i\Delta_x - x \\ j\Delta_y - y \end{pmatrix},$$

and $-\pi < \phi \leq \pi$ defines the rotation, $B(i, j)$ is the background intensity distribution, and the parameters $\sigma_{\max}$ and $\sigma_{\min}$ represent the blurring induced by the PSF and, at the same time, model the elongation of the object. For symmetrical subresolution structures such as vesicles, $\sigma_{\min} = \sigma_{\max} \approx$ 80–100 nm, and for the elongated objects, such as microtubules, $\sigma_{\min} \approx$ 80–100 nm and $\sigma_{\max} \approx$ 250–300 nm [141,161]. Concerning the density of objects in our applications, typical 512×512-pixel images contain around 50–200 objects.

2.2.2 Detection Framework

Before we describe the different detection approaches evaluated in this chapter, we first consider the detection framework in general (Fig. 2.2). This framework can be split into three subsequent steps. Each detector considered in this chapter includes these steps, but may implement them in a different way. In practice, some of the steps are optional or can be combined. Taking as input the noisy images containing the objects of interest, possibly embedded in a nonuniform background (Fig. 2.2(a)), the detector proceeds as follows:

Step 1 (Noise Reduction): The input image $\mathcal{I}$ is preprocessed using noise reduction techniques. In most cases, Gaussian smoothing [159] or matched filtering [165] is used, which may increase the SNR and improve image quality and object visibility. The output of this step is a filtered image $\mathcal{J}$ (Fig. 2.2(b)).

Step 2 (Signal Enhancement): In this step, signal processing techniques are used that enhance the denoised fluorescent light signal *only* in those regions of the image $\mathcal{J}$ where the actual objects are and, at the same time, suppress the light signal from all the background structures. That is, the image $\mathcal{J}$ is transformed to a new grayscale image $\mathcal{C}$ (Fig. 2.2(c)), also called here the grayscale classification map, which does not necessarily represent the object intensity distribution anymore. At this stage, the image $\mathcal{C}$ is rather a 2D (or 3D) signal, the value of which at any pixel measures the certainty in the object presence at that position. In other words, the image $\mathcal{C}$ can also be considered a probability map that describes possible object locations. Two examples of this classification map are shown in Fig. 2.2(d) and Fig. 2.2(e), where the image $\mathcal{C}$ in Fig. 2.2(d) is the result of applying a correlation based technique (in this case a matched filter), which convolves the image $\mathcal{J}$ with a PSF-like kernel and produces a high response in regions where objects are present (where the image intensity distribution matches the kernel), and a low response in all other image regions, suppressing the background structures. The image $\mathcal{C}$ in Fig. 2.2(e) corresponds to the situation where local background subtraction is used based on the *h*-dome transformation [177], which "cuts off" the local maxima in the image $\mathcal{J}$ in the dome-like shape of equal heights.

The described feature enhancement step does not actually detect features or objects. At this stage no quantitative information (about the object presence, its position, size, etc.) can yet be extracted and it is still up to the observer to visually link pixels that belong to one object.

Step 3 (Signal Thresholding): To obtain the number of objects and extract position information from the grayscale classification map, hard (binary) decision thresholds need to be applied. First, the image $\mathcal{C}$ is thresholded, where the threshold l_d is applied to the signal magnitude and the binary map $\mathcal{C}_B$ is obtained (Fig. 2.2(d,e)). Disjoint clusters of connected nonzero pixels in $\mathcal{C}_B$ correspond to detected objects and can be used to label the pixels in the original image $\mathcal{I}$ for subsequent analysis of the object intensity distribution. Depending on the image $\mathcal{C}$, the result of thresholding may be sensitive to the value of l_d. In that case, a second threshold $v_d = (v_{\min}, v_{\max})$ may be applied to the size and/or shape of the clusters: only those clusters in $\mathcal{C}_B$ with size larger than $v_{\min}$ and smaller than $v_{\max}$ are labeled as detected objects.

In practice, the signal thresholding with l_d does not always produce fully connected regions (clusters of pixels) in $\mathcal{C}_B$, in places where the true objects are located. In most cases, because the noise is not completely removed during Step 1, clusters of nonzero pixels in $\mathcal{C}_B$ that belong to the same spot are not connected or contain erroneous zero-pixels inside the cluster. In order to solve this problem, the closing operation from mathematical morphology [138, 149, 185] is frequently used as a post-processing step.

2.3 Detection Methods

In this section we describe the detection methods that were included in our study. All of them implement the three main steps of the general detection framework presented in the previous section. Some of the methods require noise reduction as an explicit preprocessing step to improve the detection performance, and in our analysis we include two techniques for this purpose (Gaussian filtering and wavelet denoising) that are computationally fast, easy to implement, and which are frequently used in practice (Section 2.3.1). The most characteristic feature of any detection method is its implementation of the second step of the framework (signal enhancement). As pointed out in the introduction, we make a distinction between unsupervised (Section 2.3.2) and supervised (Section 2.3.3) detection techniques. Some of them inherently reduce noise and thus do not require an explicit noise reduction step. The third step (signal thresholding) determines the final outcome of the detector, which is used to assess its performance. In the last subsection (Section 2.3.4) we describe how performance was measured in our study.

2.3.1 Noise Reduction

2.3.1.1 Gaussian Smoothing

Noise reduction in this case consists of smoothing the original image $\mathcal{I}$ with the Gaussian kernel G_σ at scale σ. The filtered image $\mathcal{J}$ is obtained as

$$J(i,j) = (G_\sigma * I)(i,j) = \sum_{i'=1}^{N_x} \sum_{j'=1}^{N_y} G_\sigma(i-i', j-j') I(i', j'), \tag{2.2}$$

where * denotes the convolution operation. (Here, and in the rest of the chapter, for all methods that require the convolution of an image with a filter kernel or mask, the image is mirrored at the borders.) In the case of additive uncorrelated noise, this smoothing can be related to matched filtering [165], which maximizes the SNR in the filtered images. This is because the PSF, which models the appearance (intensity profile) of the subcellular objects, can be approximated to a high degree of accuracy by a Gaussian [190]. The smoothed image $\mathcal{J}$ can also be used as the grayscale classification map $\mathcal{C}$, due to the fact that the image $\mathcal{J}$ is a correlation map that shows where objects similar in shape to the PSF are located. The object locations can be extracted by thresholding the image $\mathcal{J}$ in Step 3 (see Fig. 2.2), but this approach does not work in practice for typical images, which usually contain inhomogeneous backgrounds and varying object intensities.

2.3.1.2 Isotropic Undecimated Wavelet Denoising

This wavelet-based filtering technique is frequently used for image denoising in different applications [152], but also for building a separate detection procedure (Section 2.3.2.1) [52,108]. The isotropic undecimated wavelet transform (IUWT) [152,154] is well adapted to the analysis of images which contain isotropic sources, such as in astronomy [154] or in biology [52, 108], where the object appearance or shape is diffuse (no clear edges) and more or less symmetric. The denoising is accomplished by modifying the relevant wavelet coefficients and inverse transforming the result. The IUWT is usually favored over orthogonal discrete wavelet transforms (DWT) for this purpose [91]. Contrary to the DWT, the IUWT is redundant, but translation invariant, and the wavelet coefficient thresholding using an undecimated transform rather than a decimated one normally improves the result in denoising applications [151].

We used the B3-spline version of the separable 2D IUWT [108, 152], which decomposes the original image into K wavelet planes (detail images) and a smoothed image, all of the same size as the original image. The image $\mathcal{I}$ is first convolved row by row and then column by column with the 1D kernel [1/16, 1/4, 3/8, 1/4, 1/16], which is modified depending on the scale k by inserting $2^{k-1}-1$ zeros between every two taps. The image $I_{k-1}(i,j)$ is convolved with the kernel giving a smoothed image $I_k(i,j)$, and the wavelet plane is computed from these two images as

$$W_k(i,j) = I_{k-1}(i,j) - I_k(i,j), \quad 0 < k \leq K, \tag{2.3}$$

where $I_0(i,j) = I(i,j)$. Having the wavelet representation as a set of $K+1$ images, $W_1, \ldots, W_K, I_K$, also called the à trous wavelet representation, the reconstruction can be easily performed as

$$I(i,j) = I_K(i,j) + \sum_{k=1}^{K} W_k(i,j). \tag{2.4}$$

For denoising and object detection, the property of the wavelets to be localized in both space and frequency plays a major role, as it allows separation of the components of an image according to their size. The large values of $W_k(i,j)$ correspond to some structures and the smaller ones usually to noise. The denoising is based on the modification of the images $W_k(i,j)$, by hard-thresholding the coefficients, and using the modified images $\tilde{W}_k(i,j) = \mathrm{T}_d(W_k)$ in the inverse transformation (2.4). Here, the thresholding operator $\mathrm{T}_d : \mathcal{I} \rightarrow \mathcal{I}^{th}$ is defined as

$$I^{th}(i,j) = \begin{cases} I(i,j), & \text{if } |I(i,j)| \geq d, \\ 0, & \text{otherwise.} \end{cases} \tag{2.5}$$

The hard threshold d depends on the standard deviation of the wavelet coefficients σ_k per resolution level, and is usually taken to be $3\sigma_k$. Alternatively, the wavelet coefficients may be soft-thresholded according to more advanced schemes [47, 153]. However, for astronomical and also for biological images, soft thresholding should be avoided, as it leads to photometry loss in regard to all objects [153].

In order to reduce the dependence of the threshold d on the absolute values of the object and background intensities, the thresholding is often based on Bayesian analysis of the coefficient distributions using Jeffrey's noninformative prior [47] (also called the amplitude-scale-invariant), which is a nonlinear shrinkage rule that outperforms other famous shrinkage rules, including VisuShrink and SureShrink [47], and is given by

$$\tilde{W}_k(i,j) = W_k^{-1}(i,j)(W_k^2(i,j) - 3\sigma_k^2)_+, \tag{2.6}$$

where $(x)_+ = \max\{x, 0\}$. The threshold is proportional to the standard deviation of wavelet coefficients at each resolution level and it adaptively selects significant coefficients only. The modified filtered images $\tilde{W}_k(i,j)$ are used in (2.4) for the inverse transformation to obtain the denoised image $\mathcal{J}$.

2.3.2 Unsupervised Signal Enhancement

2.3.2.1 Wavelet Multiscale Product

As was mentioned in Section 2.3.1.2, in the à trous wavelet representation, contrary to the frequently used orthogonal wavelet transform [91], the wavelet coefficients are correlated across the resolution levels (scales). This property is exploited by the detection approach based on the multiscale product [108], which uses the same image decomposition as in Section 2.3.1.2 and creates the multiscale product image as

$$P_K(i,j) = \prod_{k=1}^{K} W_k(i,j). \tag{2.7}$$

This transformation constitutes Step 2 in the general detection framework (Section 2.2.2). For better performance, the original algorithm [108] also includes the noise reduction step (Step 1) using the technique described in Section 2.3.1.2: the wavelet coefficients are hard-thresholded per scale, $\tilde{W}_k(i,j) = \mathrm{T}_{d_k}(W_k(i,j))$, with the threshold $d_k = k_d\sigma_k$, $k_d = 3$, and the modified coefficients $\tilde{W}_k(i,j)$ are used in (2.7).

This method uses the fact that the real objects are represented by a small number of wavelet coefficients that are correlated across the scales. Contrarily, the coefficients that are due to noise are randomly distributed and are not propagated across scales. As a result, the image $P_K(i,j)$, which is the grayscale classification map $\mathcal{C}$, is thresholded with l_d and binarized. The connected components in the binary map $\mathcal{C}_B$ are considered as detected objects (Step 3). In the original algorithm [108], $l_d = 1.0$ and no thresholds on the cluster size v_d in the thresholded and binarized $P_K(i, j)$ were imposed [108]. In summary, this method has three parameters, (l_d, k_d, K), that are not directly related to the object appearance. Recently, a modification of the described method, which uses the Gaussian kernel at several scales instead of B3-splines, was proposed for segmentation and analysis of nuclear components in stem cells [176].

2.3.2.2 Top-Hat Filter

Another class of methods that are used for detection of bright spots in the presence of widely varying background intensities is known as top-hat filters [20, 21]. Such filters are dynamic thresholding operators, rather than the similarly named image transformation from mathematical morphology. The latter transformation selects extended objects with sufficiently narrow parts, rather than compact objects, as does the top-hat filter considered here.

The filter discriminates the spots by their round shape and predetermined information about their intensity and size. At each pixel location, (i, j), the average image intensitiy $\bar{I}_{\text{top}}$ and $\bar{I}_{\text{brim}}$ are calculated for pixels within two circular regions D_{top} and D_{brim}, respectively, defined as

$$D^{i,j}_{\text{top}} = \{(i', j') : (i - i')^2 + (j - j')^2 < R^2_{\text{top}}\}, \tag{2.8}$$

$$D^{i,j}_{\text{brim}} = \{(i', j') : R^2_{\text{top}} < (i - i')^2 + (j - j')^2 < R^2_{\text{brim}}\}, \tag{2.9}$$

where the radius R_{top} corresponds to the "top" of the "hat" and is set to the maximum expected spot radius. The brim radius, R_{brim} ($R_{\text{brim}} > R_{\text{top}}$), is often taken to be the shortest expected distance to the neighboring spot. If the difference $\bar{I}_{\text{top}} - \bar{I}_{\text{brim}}$ is larger than some threshold H_{th}, the original image intensity $I(i, j)$ for that position (i, j) is copied to the classification map $\mathcal{C}$, $C(i, j) = I(i, j)$, otherwise $C(i, j) = 0$. The procedure is repeated for each pixel, and the binary map $\mathcal{C}_B$ (Step 3) is obtained as $C_B(i, j) = 1$ if $C(i, j) \neq 0$, and $C_B(i, j) = 0$ otherwise. The connected components are counted without any size or shape threshold.

The height H_{th} of the top above the brim is set to the minimum intensity that a spot must rise above its immediate background. It can also be related to the minimum local SNR that we are willing to deal with. If the detection of all the objects with local SNR $> a$ is required, because for lower SNRs the detector would produce a lot more false positives and contaminate the analysis, the threshold H_{th} can be fixed to $a\sigma_{\text{brim}}$, where σ_{brim} is the standard deviation of the intensity distribution in the region D_{brim}.

In summary, the described algorithm has only three parameters, $(H_{th}, R_{\text{top}}, R_{\text{brim}})$, which can be related to the object appearance. The noise reduction (Step 1) in this case is implicitly done while calculating the average image intensitiy $\bar{I}_{\text{top}}$ and $\bar{I}_{\text{brim}}$. The averaging decreases the variance in the estimation of the noisy object and background intensity levels and improves the robustness and performance of the method. A slightly modified version of the filter, called the top-hat box filter [20], uses a square mask for the regions D_{top} and D_{brim} and is computationally faster, but in the present context this is not an important advantage.

2.3.2.3 Spot-Enhancing Filter

The optimal filter for enhancing subresolution particles and reducing correlated noise in microscopy images is the whitened matched filter, which is well approximated by the Laplacian of a Gaussian (LoG) [128]. In this case, the convolution kernel $(2\sigma_L^2 - i^2 - j^2)\sigma_L^{-4}G_{\sigma_L}$ is used in (2.2) to obtain the image $\mathcal{J}$, where the filter parameter σ_L must be tuned to the size of the particles. The filter combines Steps 1 and 2 and operates as a local background subtraction technique that preserves object-like structures and removes the background and noise. The filter can be made computationally fast by separable implementation [128]. The result of LoG filtering, the image $\mathcal{J}$, is used as the classification map $\mathcal{C}$, which is thresholded with l_d to locate the objects. This detection procedure has two parameters, $(\sigma_L,\, l_d)$, and is similar to the top-hat filter (Section 2.3.2.2), with the difference that here the convolution kernel, also called the "Mexican hat", represents a continuous version of the top-hat filter mask.

2.3.2.4 Grayscale Opening Top-Hat Filter

Similar to the method above (Section 2.3.2.2), this top-hat filter uses the opening operation from mathematical morphology [138, 147, 149]. In order to improve the detector performance, the original image $\mathcal{I}$ is first smoothed with the Gaussian kernel with scale σ (Step 1) and the grayscale opening of $\mathcal{J}$ with a structuring element A is done, producing the image $\mathcal{J}_A$, where in our case a flat disk of radius r_A is used. The radius r_A is related to the size of the largest objects that we would like to detect. The top-hats are obtained after the subtraction $\mathcal{C} = \mathcal{J} - \mathcal{J}_A$ (which concludes Step 2), and the whole transformation acts as a background subtraction method that leaves only compact structures smaller than the disk A, or extended objects with sufficiently narrow parts, rather than compact objects only, as does the top-hat filter. The resulting image $\mathcal{C}$ is thresholded at level l_d (Step 3), and then all the connected components are counted. Additional filtering with v_d can be done if the size of the connected components should be taken into account. Thus, this method has four parameters, $(\sigma,\, r_A,\, l_d,\, v_d)$, all of which can be related to the object appearance.

2.3.2.5 H-Dome Based Detection

Another approach borrowed from grayscale mathematical morphology is based on the h-dome transformation [177], which was used in our previous works on subresolution particle tracking to design a detection scheme for track initiation and termination [142, 146]. The transformation has the interesting property that all the detected objects end up having the same maximum intensity in the transformed image, which we exploited to build a fast probabilistic tracker that outperforms current deterministic methods [146] and at the same time has the same tracking accuracy as the computationally more expensive particle filtering approaches for tracking [141, 146].

For this method we assume that the intensity distribution in the image $\mathcal{I}$ is formed by N_o objects (bright spots), modeled using (2.1), background structures (also called clutter) with intensity distribution $B(i, j)$, and possibly spatially correlated additive or multiplicative noise $\eta(i, j)$. The main problem is to accurately estimate the number of real objects N_o and the object positions $(x_l, y_l)^T$, $l = \{1, \ldots, N_o\}$, in the presence of inhomogeneous background structures and noise. The algorithm also consists of three steps: filtering, h-dome transformation, and “sampling” (signal thresholding). First, the image $\mathcal{I}$ is LoG filtered with scale σ_L, which enhances the signal in the places where objects are present and performs local background subtraction (Step 1). The scale σ_L can be related to the size of the objects to be detected, and in our experiments is equal to 2.5 pixels (125 nm). Then, grayscale reconstruction [177] is performed on the LoG-filtered image $\mathcal{J}$ with mask image $\mathcal{J} - h$, where $h > 0$ is a constant (Step 2). As a result, the original image is decomposed into the reconstructed image $\mathcal{B}_\sigma$ and the so-called h-dome image $\mathcal{H}_\sigma$:

$$\mathcal{I}_\sigma(i,j) = \mathcal{H}_\sigma(i,j) + \mathcal{B}_\sigma(i,j). \tag{2.10}$$

Geometrically speaking, similar to local background subtraction, the h-dome transformation extracts bright structures by “cutting off” the intensities of height h from the top, around local intensity maxima, producing “dome”-like structures. Contrary to top-hat filtering [177], this does not involve any shape or size criteria. The image $\mathcal{B}_\sigma$ represents the nonuniform background structures, and image $\mathcal{H}_\sigma$ contains the objects and all the smaller noise structures.

After the transformation, the maximum intensity of those Gaussian-like objects is approximately h, and for the noise structures the amplitude is less than h [146]. This transformed image $\mathcal{H}_\sigma$ is used as a probability map for the final step of the algorithm (Step 3): the sampling. During this step, all the pixel values in $\mathcal{H}_\sigma$ are raised to the power s in order to compensate for the broadening of the original object intensity distributions by the convolution with the LoG filter, and to create a highly peaked function that resembles the probability density function (pdf) of the object location distribution. The parameter s can be related to the maximum and minimum object size and the scale σ_L [146]. The function $H_\sigma^s(i,j) = (J(i,j) - B_\sigma(i,j))^s$ is used in our framework as a so-called importance sampling function [9], denoted by $q(i,j|\mathcal{I})$, that describes which areas of the image most likely contain the objects. We sample N position-samples from $q(i,j|\mathcal{I})$ using systematic resampling [9], $\mathbf{x}^l \sim q(i,j|\mathcal{I})$, where $l = \{1, \ldots, N\}$ and $\mathbf{x} = (i,j)$, in order to estimate the object positions using Monte Carlo methods. Then, the mean-shift algorithm [34] is used to cluster the samples $\mathbf{x}^l$, resulting in M clusters. For each cluster, the mean position $\mathbf{x}_c = (i_c, j_c)$ and the variance R_c are computed using only the N_c samples $\mathbf{x}^l$ belonging to that cluster:

$$\begin{aligned} \mathbf{x}_c &= \mathbb{E}[\mathbf{x}_c^l] = N_c^{-1} \textstyle\sum_{l=1}^{N_c} \mathbf{x}_c^i, \\ R_c &= \mathbb{E}[(\mathbf{x}_c^l - \mathbf{x}_c)(\mathbf{x}_c^l - \mathbf{x}_c)^T]. \end{aligned} \tag{2.11}$$

The following two criteria are used to distinguish between real objects and other structures: 1) the number of samples N_c in the cluster should be larger than the number of samples in case of sampling from the uniform intensity distribution in the image region occupied by the cluster, and 2) the determinant of the covariance matrix of the cluster, $\det R_c$, must be less than $\sigma_{\mathrm{M}}^4/s^2$, where σ_{M} characterizes the maximum object size that we are interested in. These two thresholds are motivated by the fact that the elements of the estimated covariance matrix R_c using the samples generated from the intensity distribution of the real objects, are bounded from above by $(\sigma_{\max}^2 +$

$\sigma_L^2)s^{-1}$. The samples that came from noise have approximately the same variance ($R_c \approx \sigma_L^2 \mathbf{I} s^{-1}$), where $\mathbf{I}$ is the identity matrix, but since the intensity amplitude $\ll h$, the number of samples N_c in the corresponding cluster will be below the mentioned threshold. The clutter on the other hand, having possibly high intensity values ($\approx h$), produces a large number of samples, but the variance in those clusters is higher than in the case of the largest real object characterized by σ_{M}.

The parameters σ_L and σ_M of this detection method can be related to the object appearance. The height h is related to the SNR in the same way as in the case of the top-hat filter (Section 2.3.2.2). The method is fairly insensitive to the free parameters s and N [142,146] (above some minimum, sensible values, which can be found experimentally and then fixed, these parameters primarily affect the computational cost of the method, not its accuracy). Thus, in summary, this method depends mainly on three parameters, (σ_L, σ_M, h), that need to be tuned to the application.

2.3.2.6 Image Features Based Detection

The last unsupervised method that we consider in this study is based on using some additional image information during Step 2 that would help to distinguish the spots from the clutter. As was shown previously [141,161], the incorporation of local curvature information can be used to build a reasonably good detector for image data with SNR > 4. The true spots in the image are characterized by a combination of convex intensity distributions and a relatively high intensity. Noise-induced local maxima typically exhibit a random distribution of intensity changes in all directions, leading to a low local curvature [161]. These two discriminative features (intensity and curvature) are used in combination during Step 2 to create the grayscale classification map $\mathcal{C}$ using the denoised image (Step 1) $J(i,j) = (G_\sigma * I)(i,j)$ as follows:

$$C(i,j) = J(i,j)\kappa(i,j), \tag{2.12}$$

where the curvature $\kappa(i,j)$ at each pixel of $\mathcal{J}$ is given by the determinant of the Hessian matrix $\mathbf{H}(i,j)$ [159], which itself is known to be a good blob detector [86]. The classification map $\mathcal{C}$ again is binarized (Step 3) using the threshold l_d and possibly the size threshold v_d which are not directly related to the object appearance.

2.3.3 Supervised Signal Enhancement

In order to make our comparison study of spot detection methods more complete, we also included two machine learning (ML) techniques. The first one is the AdaBoost algorithm [178], which is frequently used for object detection in computer vision [50, 85,178], and was recently shown to perform well also for spot detection in molecular bioimaging [73]. The second method is Fisher discriminant analysis (FDA), which is a classical and well-known linear classifier, but which has not been employed for spot detection in fluorescence microscopy up to now. It uses the same information as AdaBoost but is computationally less expensive and much easier to understand conceptually.

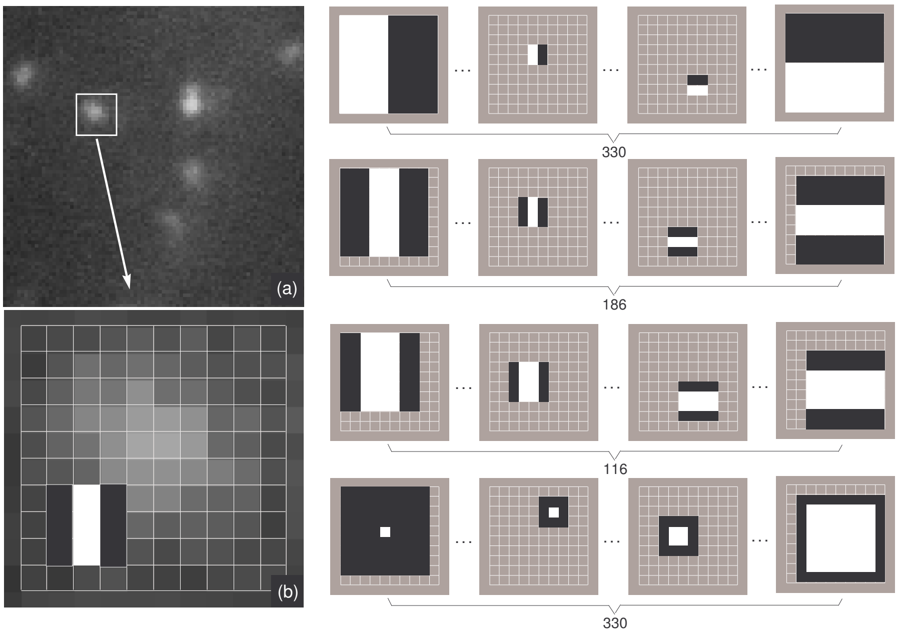

Figure 2.3. Examples of the Haar-like features that were used in the experiments to detect spots, and the numbers of all possible scaled and translated versions in 10×10-pixel subwindows of the image.

2.3.3.1 AdaBoost

This ML detection algorithm operates on small patches of the image around the hypothesized spot positions (Fig. 2.3(a)) and classifies the patches (Fig. 2.3(b)) as positive (object is present) or negative (object is absent) based on the combined response of several simple feature-based classifiers. Usually the feature-based systems are favored over pixel-based ones because they are much faster and can encode some domain knowledge. A set of N_F simple Haar-like features is used [111], which is overcomplete in comparison with the real Haar basis [91], and in our case consists of four kinds (four different rows in Fig. 2.3). For each feature η_l, $l = \{1, \ldots, N_F\}$, the feature value $\xi(\eta_l)$ is a weighted difference between the sum of the pixels within two (black and white) rectangular regions. The weights are chosen in such a way that the value of the feature computed for constant-intensity images is zero. The number of possible features, which are scaled and translated versions of the features of each kind (Fig. 2.3), depends on the image patch size, and for 10×10-pixel image subwindows [73] is 962 (the number of features per kind is indicated below each feature row in Fig. 2.3). Using the integral images [178], the computation of the sums of pixels in the rectangular regions can be performed very fast.

Having the pool of N_F features η_l, and a training set consisting of N_T image patches labeled as positive and N_T patches labeled as negative, we selected a variant of the AdaBoost learning algorithm that can be used both to select a small subset of features and to train the classifier [178]. Such a choice was made on the basis of recently published results of applying the AdaBoost algorithm in bioimaging [73]. The AdaBoost algorithm is used to boost the performance of a simple (weak) learning algorithm. The weak classifier is designed to select the single feature that best separates the positive and negative samples. In our case, this separation is accomplished by finding the appropriate threshold d_l for each feature η_l at every round during the training stage. With each run of the algorithm, one feature is selected and added to the set of best discriminating features. The number of runs, denoted by N_{AB}, is user-defined. It is known that the training error of the strong classifier approaches zero exponentially in the number of rounds [50].

The final strong classifier is a weighted linear combination of all selected weak classifiers. The classification map $\mathcal{C}_B$ (Step 3) is constructed as follows. First, for each pixel (i, j) of $\mathcal{I}$ the value of the feature $\eta_{l'}$ is computed using the corresponding 10×10-pixel image subwindow centered at (i, j) and assigned to $C^{l'}(i, j)$, where l' specifies one of the N_{AB} features that were selected during the training. This way, the image $\mathcal{C}^{l'}$ is obtained. Then, the values in $\mathcal{C}^{l'}$ are thresholded using the feature threshold $d_{l'}$, producing a binary version $\mathcal{C}^{l'}_B$ of $\mathcal{C}^{l'}$. The procedure is repeated for all N_{AB} features, and the images $\mathcal{C}^{l'}_B$, $l' = 1, \ldots, N_{AB}$, are combined (with weights also learned during the training) into $\mathcal{C}$, which is then thresholded with the threshold $l_d = 0.5$ [178], producing the map $\mathcal{C}_B$. In the final classification map, some additional thresholding using the size information v_d (not related to the notion of spot size) might be needed in order to remove small regions with misclassified pixels.

By applying the trained classifier to the image $\mathcal{I}$ (Step 2), prefiltering (Step 1) is performed implicitly: the values of the features are the difference in average pixel values in the black and white rectangular regions. This averaging reduces the variance of the feature value estimation in a similar way as in the case of the top-hat filter (Section 2.3.2.2).

2.3.3.2 Fisher Discriminant Analysis

Discriminant analysis is a statistical technique which classifies objects into one of two or more groups based on a set of features that describe the objects [93]. We use FDA to classify the image patches in the same way as in the AdaBoost method (Section 2.3.3.1). For an image patch of size $n \times n$ pixels, the n horizontal rows of pixels are concatenated into a 1-D (column) feature vector $\mathbf{y}$ of size n^2. Having a labeled training dataset with positive and negative samples (image patches), the corresponding sets of features $\{\mathbf{y}_1^l\}_{l=1}^{N_T}$ and $\{\mathbf{y}_0^l\}_{l=1}^{N_T}$ are used to compute the mean $\boldsymbol{\mu}_c$ and the covariance matrix $\boldsymbol{\Sigma}_c$ for each class $c = \{0, 1\}$. The task of FDA is to find the linear transformation $\mathbf{w}$ that maximizes the ratio

$$Q(\mathbf{w}) = \frac{(\mathbf{w}^T(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0))^2}{\mathbf{w}^T(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_0)\mathbf{w}}. \tag{2.13}$$

In some sense, $Q(\mathbf{w})$ is a measure of the SNR for the class labeling, where the numerator represents the between-class variation and the denominator represents the within-class variation. It can be shown that the optimal separation occurs when $\mathbf{w} = (\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_0)^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$ [93]. This concludes the training stage. During the classification stage, when FDA is applied to patches extracted from the image $\mathcal{I}$ using a sliding subwindow of size $n \times n$ pixels, the patch is classified as positive (object is present, $C_B(i,j) = 1$) if the condition $|\mathbf{w}^T\mathbf{y} - \boldsymbol{\mu}_1| < |\mathbf{w}^T\mathbf{y} - \boldsymbol{\mu}_0|$ is satisfied, and as negative (object is absent, $C_B(i,j) = 0$) otherwise.

The FDA classification procedure has an appealing interpretation as linear filtering (similar to (2.2)) with a kernel that is learned from the training data. The n^2-dimensional vector $\mathbf{w}$ can be reshaped into an $n \times n$ patch, similar to the image patch from which the feature vector $\mathbf{y}$ is formed (see examples in Section 2.4.2, Fig. 2.17). In this case, the projection $\mathbf{w}^T\mathbf{y}$, which is performed using the sliding subwindow for each image pixel, is a convolution as in (2.2). The classification map $\mathcal{C}$ is obtained by thresholding the convolution result at $l_d = \frac{1}{2}\mathbf{w}^t(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$, which is obtained automatically because the training was performed beforehand.

2.3.4 Signal Thresholding and Performance Measures

As mentioned before, in order to locate and count the detected objects, the classification map $\mathcal{C}$ is binarized using the threshold l_d (whose meaning depends on the method), and the connected components are searched for. Having the binary image $\mathcal{C}_B$, where $C_B(i,j) = 1$ if $C(i,j) > l_d$, and $C_B(i,j) = 0$ otherwise, we run the sequential scan labeling algorithm [66] in order to label the connected components and obtain the set of labels $L(i,j)$ for all pixels, where $L(i,j) \in \{0,\ldots,M\}$, with $L = 0$ corresponding to the background and $L \neq 0$ denoting one of the M detected objects. The center of mass, $\mathbf{x}_m$, is calculated for each of M objects, taking into account the pixels (i,j) and the image intensity $I(i,j)$ for all (i,j) for which $L(i,j) = m$. The position is compared to the "ground truth" $\mathbf{x}_m^0$ (known exactly in the case of synthetic images, and obtained manually by approximation in the case of real biological images). If $\|\mathbf{x}_m^0 - \mathbf{x}_m\| < \Delta^0$, the object is counted as a true positive (TP), otherwise the detected object is a false positive (FP). The number of false negatives (FN) is defined as $N^0 - N_{\mathrm{TP}}$, where N^0 is the number of objects in the ground truth and N_{TP} is the number of TPs. True negative (TN) is defined as accurate detection of the spot not to be an object. The number of TNs can be defined only for the ML approaches during the training stage. During the actual detection with any of the described methods, the number of TNs in the image data is undefined.

In order to measure the performance of the algorithms, we consider two common measures: the true-positive ratio (TPR), TPR $= N_{\mathrm{TP}}/(N_{\mathrm{TP}} + N_{\mathrm{FN}}) = N_{\mathrm{TP}}/N^0$, also called sensitivity, and the false-positive ratio (FPR), FPR $=N_{\mathrm{FP}}/(N_{\mathrm{FP}} + N_{\mathrm{TN}})$. Because TN is not known for some methods, the modified version of FPR is used, given by FPR*=N_{FP}/N^0. In this case, the standard receiver operating characteristic (ROC) curve cannot be built, and the modified version, called the free-response receiver operating characteristic (FROC) curve, is used [29,30]. To demonstrate the sensitivity of TPR and FPR* to parameters, for example the threshold l_d, we measure the values $S_T = -(\partial\mathrm{TPR}/\partial l_d)$ and $S_F = -(\partial\mathrm{FPR}^*/\partial l_d)$ at $l_d = l_d^*$. The threshold l_d^* is hereafter called "optimal" and corresponds to the value for which the FPR$^* = 0.01$ (only 1% false positives). The value of TPR for $l_d = l_d^*$ is denoted as TPR*. Having S_T and S_F, we can compute the value $\Delta\mathrm{TPR} = 0.01 S_T l^*$, which corresponds to the changes in TPR (around TPR*) when the parameter value l_d (or v_d) is changed by 1% around l^* (or v^*). Similarly, $\Delta\mathrm{FPR} = 0.01 S_F l^*$ can be introduced for the FPR.

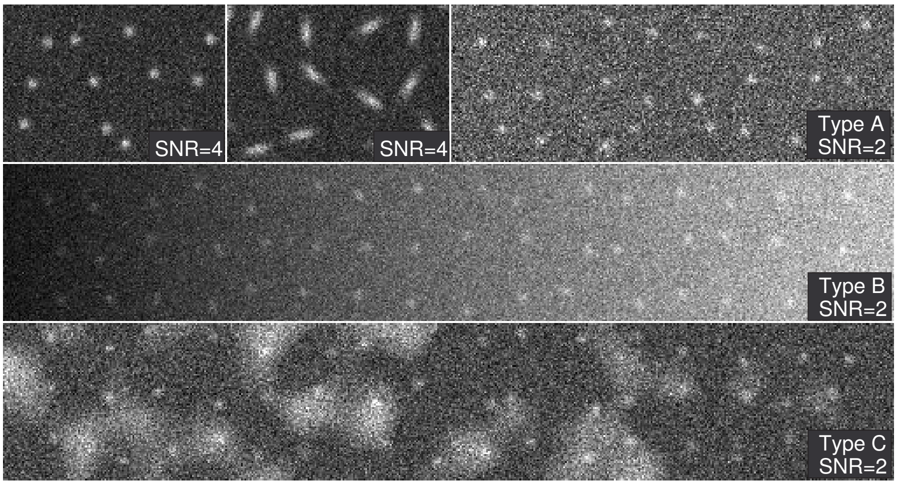

Figure 2.4. Examples of synthetic images used in the experiments. The symmetrical Gaussian intensity profiles are embedded into uniform (Type A), gradient (Type B), and non-uniform (Type C) backgrounds.

2.4 Experimental results

The performance of the eight detection methods (six unsupervised and two supervised methods) described in the previous section was quantitatively evaluated using both synthetic images (Section 2.4.1) and real image data (Section 2.4.2) acquired for different biological studies. In the experiments, we studied the dependence of the performance (TPR and FPR*) on parameter settings, type of object (perfectly round or slightly elongated), and image quality (SNR). Here we describe the experimental setups and the results.

2.4.1 Evaluation on Synthetic Image Data

2.4.1.1 Simulation Setup

The described detection methods were evaluated using synthetic but realistic 2D images (of size 512×512 pixels, with $\Delta_x = \Delta_y = 50$ nm) containing intensity profiles of round and elongated objects modeled using (2.1) with $\sigma_{\max} = \sigma_{\min} = 100$ nm for round objects, and $\sigma_{\max} = 250$ nm, $\sigma_{\min} = 100$ nm for elongated objects, for different levels of Poisson noise in the range of SNR = 2–4. Such SNRs are typical for the real image data acquired in our biological applications and are lower than the critical level of SNR = 4–5, at which several classical detection methods break down [26,32]. Here, SNR is defined as the difference in intensity between the object and the background, divided by the standard deviation of the object noise [32].

In order to estimate the performance of the algorithms, three types of images were created (see Fig. 2.4), for each type of object shape and for each SNR. In every image, 256 Gaussian intensity profiles were placed at positions $\mathbf{x}^0_{i',j'} = (16 + 30i' +$

$\mathcal{U}_{[-10,10]}, 16{+}30j'{+}\mathcal{U}_{[-10,10]})^T$, where $i' = 0, \ldots, 15$, $j' = 0, \ldots, 15$, and $\mathcal{U}_{[-a,a]}$ denotes the uniform random generator within the interval $[-a, a]$. This way, the objects were randomly placed, with no overlaps in the intensity distributions. Type A images were constructed by adding a background level of 10, similar to previous studies [32]. To form the final noisy image, a Poisson noise generator was applied independently to every pixel of the noise-free image. In the case of Type B images, the background level increased linearly in the horizontal direction (see Fig. 2.4), from a value of 10 at the left image border to 50 at the right border. Taking into account that the variance of Poisson noise is intensity dependent, we corrected the object intensities accordingly prior to application of the noise generator in order to keep the SNR constant over the whole image. Finally, type C images mimic the intensity distribution in the presence of large (compared to object size) background structures (clutter), which are sometimes present in the real image data and can be either larger subcellular structures or acquisition artifacts. In this case, the pixel values were sampled from the normal distribution $I_0(i,j) \sim \mathcal{N}(0, 150)$. Then, the image was convolved with the Gaussian kernel G_{10} and thresholded at zero-level. The final image $\mathcal{I}$ was obtained by adding to $\mathrm{T}_0(G_{10} * I_0)$ a constant background level of 10 plus the (SNR-adapted) object intensity profiles, followed by application of Poisson noise. Examples of synthetic images of all three types are shown in Fig. 2.4. In every experiment, the performance of the detection techniques for each object type was evaluated by accumulating the numbers of TP and FN for 16 images (each containing 256 ground truth objects) and averaging the results over the 4096 objects. The distance between the ground truth location and the object position estimated by the detector, Δ^0, which defines if the detected object is a TP or FP, was fixed to $\Delta^0 = 200$ nm (4 pixels).

2.4.1.2 Wavelet Multiscale Product

For the performance evaluation of the wavelet multiscale product detector (further abbreviated as WMP), the parameters of the method (see Section 2.3.2.1) were fixed to the values described in the original paper [108]: $l_d = 1$, $K = 3$, $k_d = 3$. The performance measures TPR and FPR* for the image data with SNR = 2 are shown in Table 2.1. In order to evaluate the sensitivity of the method to parameter changes, we varied the number of scales K and the wavelet coefficient threshold k_d in our experiments and studied their influence on the behavior of TPR and FPR*. In the experiments, the grayscale classification map $\mathcal{C}$ produced by the method was thresholded at l_d, and after binarization all the connected components were labeled as detected objects. Because the method produced quite fractured clusters of pixels, we used the morphological opening operator with a square 3×3 mask (a 5×5 mask yielded very similar results) in order to fill in the holes.

The main results of the sensitivity analysis for this method are shown in Fig. 2.5. They show that a value of $K = 3$ is a good compromise to maximize performance for all three different data types together (Fig. 2.5(a)-(c)). The results also show that the performance of this method drops quite rapidly when the SNR decreases from 4 to 2 (Fig. 2.5(d)), and also when the background complexity increases (Fig. 2.5(e)-(f)). Table 2.2 shows the “optimal” values of k_d for different types of data for $l_d = 1$, $K = 3$, and SNR = 2.

Table 2.1. Performance of the WMP detector using the original algorithm parameters at SNR = 2.

Image	Round Objects		Elongated Objects	
Type	TPR	FPR*	TPR	FPR*
A	0.33	0.001	0.34	0.013
B	0.18	0.001	0.20	0.010
C	0.21	0.015	0.25	0.017

Table 2.2. Optimal parameters and performance of the WMP detector at SNR = 2 and number of scales $K = 3$.

Image	Round Objects				Elongated Objects			
Type	k_d^*	TPR*	S_T	S_F	k_d^*	TPR*	S_T	S_F
A	2.22	0.81	.57	.04	3.06	.31	.61	.05
B	2.56	0.37	.56	.05	3.07	.17	.36	.05
C	2.89	0.30	.62	.09	3.17	.18	.39	.06

For comparison, we also applied the soft thresholding of the wavelet coefficients according to (2.6) instead of the original hard thresholding with $k_d = 3$. For round objects in Type C images at SNR = 2, using the hard threshold $k_d = 3$, we had FPR* = 0.015 and TPR = 0.21. The value of l_d was increased to 34 when the soft threshold (2.6) was used in order to obtain the same FPR*, and the TPR in this case was equal to 0.25. For elongated objects the corresponding values were FPR* = 0.017 and TPR = 0.25 for the hard thresholding, and TPR = 0.27 for the soft thresholding.

Another experiment was conducted in order to investigate if the low performance of the WMP for SNRs around 2–3 was dependent on the type of noise (Poisson versus Gaussian). The variance-stabilizing Anscombe transform [7] was applied, which transforms the image intensities according to $I(i,j) \rightarrow 2\sqrt{I(i,j) + 3/8}$, and creates approximately Gaussian data of unit variance, provided that the mean value of the Poissonian data is more than 10 [7]. The experiments with the variance-stabilized (Gaussian) images showed no significant difference in TPR and FPR for all types of image data compared to the original (Poissonian) synthetic images.

2.4.1.3 Top-Hat Filter

The performance of the top-hat filter (further abbreviated as TH) was evaluated using the same images as for the WMP detector. The brim radius, R_{brim}, which controls the local background estimation around the spot position, was fixed to 10 (see Section 2.3.2.2 for the parameters description). Varying this parameter in the range 8-12 did not influence the final results significantly, indicating that the local background estimation is quite robust. The TPR and FPR* of the method for different R_{top} values, depending on H_{th}, are shown in Fig. 2.6. Again, holes within clusters (objects) in the binarized classification map $\mathcal{C}_B$ were filled using the closing operation with a 5×5 mask. All found clusters were considered as objects, regardless of cluster

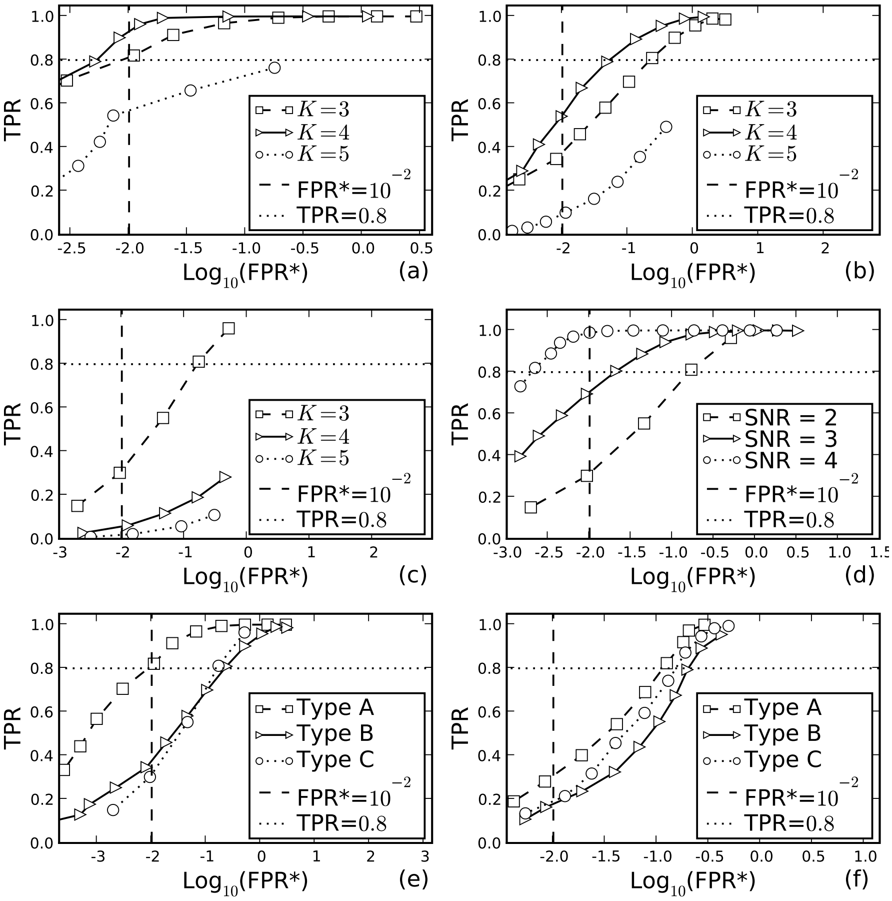

Figure 2.5. FROC curves for the WMP detector in the case of the round objects, depending on the wavelet coefficient threshold k_d, for Type A (a), Type B (b), and Type C (c) image data and different numbers of scales K, and the FROC curves for Type C data for different SNRs (d). The same type of FROC curves in the case of the round (e) and elongated (f) objects for different types of data, with SNR = 2 and $K = 3$.

size. The optimal values of H_{th} for all image types with SNR = 2 are shown in Table 2.3. The value of $R_{\text{top}} = 3$ was chosen, which maximizes the TPR when FPR$^* = 0.01$ for Type C data with both round and elongated objects.

2.4.1.4 Spot-Enhancing Filter

The performance of the spot-enhancing filter (further abbreviated as SEF) using the synthetic images was studied depending on the values of the signal threshold l_d (see Section 2.3.2.3). The filter acts as a smoothing and local background subtraction technique at the same time (Steps 2 and 3). The only parameter is the scale of

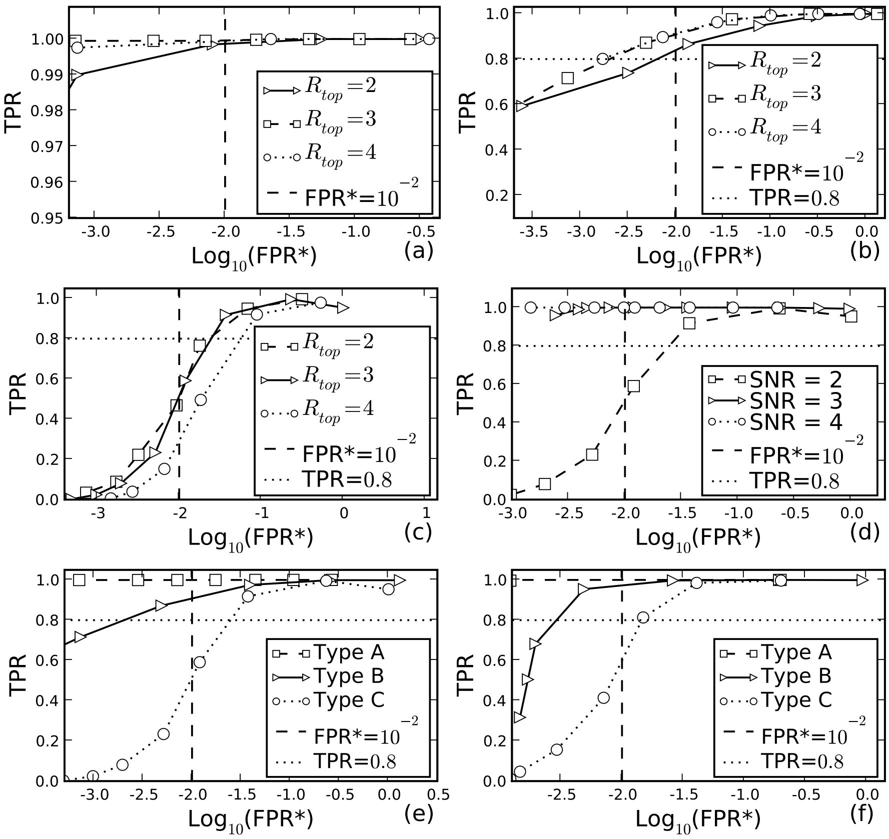

Figure 2.6. FROC curves for the TH detector in the case of the round objects, depending on the values of H_{th}, for several values of R_{top}, for Type A (a), Type B (b), and Type C (c) image data, and the FROC curves for Type C data for several SNRs (d). The same type of FROC curves in the case of the round (e) and elongated (f) objects depending on the values of H_{th} for different types of data, with SNR = 2, $R_{\mathrm{brim}} = 10$, and $R_{\mathrm{top}} = 3$.

Table 2.3. Optimal parameters and performance of the TH detector at SNR = 2 with radii $R_{\mathrm{brim}} = 10$ and $R_{\mathrm{top}} = 3$.

Image	Round Objects				Elongated Objects			
Type	H_{th}^*	TPR*	S_T	S_F	H_{th}^*	TPR*	S_T	S_F
A	2.74	.99	.00	.05	2.95	.99	.00	.20
B	5.85	.88	.11	.03	5.75	.96	.04	.02
C	5.28	.48	.35	.01	5.62	.56	.38	.01

the convolution kernel, σ_L, which was tuned in order to get the highest TPR at FPR$^* = 0.01$ in the case of Type C data. In the case of round objects, for σ_L values

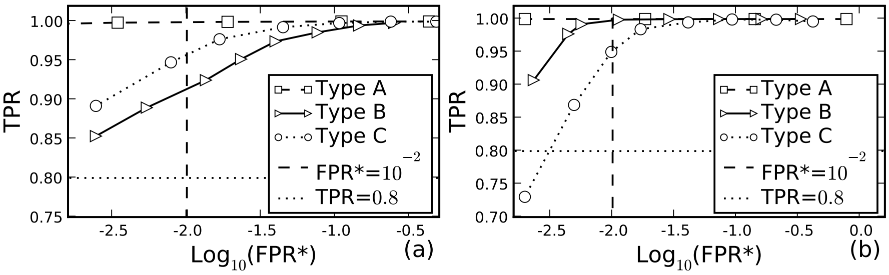

Figure 2.7. FROC curves for the SEF detector in the case of round (a) and elongated (b) objects, depending on the values of the threshold H_{th} and the type of image data, at SNR = 2 and optimal scales $\sigma_L = 2.5$ (for round objects) and $\sigma_L = 3.1$ (for elongated objects).

Table 2.4. Optimal parameters and performance for the SEF detector at SNR = 2 and optimal scales $\sigma_L = 2.5$ (for round objects) and $\sigma_L = 3.1$ (for elongated objects).

Image	Round Objects				Elongated Objects			
Type	l_d^*	TPR*	S_T	S_F	l_d^*	TPR*	S_T	S_F
A	0.85	.99	.01	.15	0.55	.99	.00	.16
B	1.84	.91	.35	.08	1.21	.99	.07	.06
C	1.22	.95	.29	.09	0.99	.95	.34	.07

$\{1.5, 2, 2.5, 3, 3.5\}$, the corresponding TPR values were $\{0.52, 0.9, 0.95, 0.9, 0.65\}$, and thus $\sigma_L = 2.5$ was used in the experiments. In the case of elongated objects, for σ_L in $\{2.5, 3, 3.5, 4\}$, the corresponding TPR values were $\{0.75, 0.86, 0.92, 0.74\}$, and $\sigma_L = 3.1$ was used. All clusters in the binary classification map after signal thresholding were counted as objects, and the values l_d^* and corresponding TPR*, S_T, and S_F, for which FPR* = 0.01, are shown in Fig. 2.7 and Table 2.4. Again, the value l_d^* represents the optimal threshold, for which FPR* = 0.01, with corresponding TPR denoted as TPR*.

2.4.1.5 Grayscale Opening Top-Hat Filter

This detection method from grayscale morphology (further abbreviated as MTH) is a robust local background subtraction technique. Its performance was not influenced significantly by changes of the mask size, r_A, in the range $(3, 5)$ (see the parameter description in Section 2.3.2.4). The input images were first smoothed with the Gaussian kernel at $\sigma = 2$. The radius of the mask was fixed to $r_A = 5$, which means that all image structures of size smaller than the size of the disk A would be translated to the detection map $\mathcal{C}$. Two thresholds, one on the intensity amplitude and one on the object size, could be applied for the object extraction from $\mathcal{C}$. The latter threshold is crucial if the clutter consists of possibly elongated narrow structures, which would be considered as objects by this detector (see Section 2.3.2.4). We studied the dependence of TPR and FPR* only on the intensity threshold l_d, as in the synthetic images

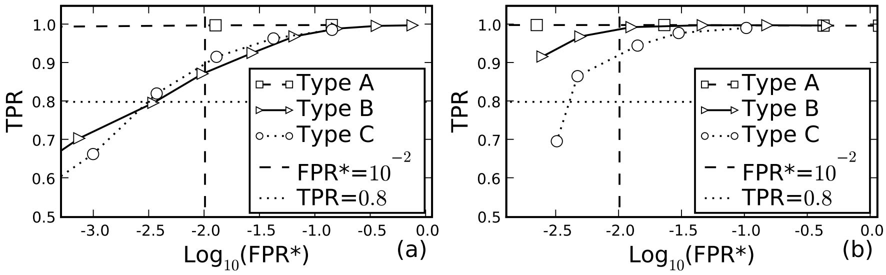

Figure 2.8. FROC curves for the MTH detector in the case of round (a) and elongated (b) objects, depending on the values of intensity threshold l_d for different types of image data, at SNR = 2, and with mask radius r_A = 5 and Gaussian prefiltering at σ = 100 nm.

Table 2.5. Optimal parameters and performance for the MTH detector at SNR = 2 and with mask radius r_A = 5 and Gaussian prefiltering at σ = 100 nm.

Image	Round Objects				Elongated Objects			
Type	l_d^*	TPR*	S_T	S_F	l_d^*	TPR*	S_T	S_F
A	2.1	.99	.00	.04	2.1	.99	.00	.04
B	3.5	.87	.18	.06	4.1	.98	.05	.02
C	2.2	.88	.31	.03	3.2	.91	.15	.02

there are no clutter structures smaller than the object size. In this case, either intensity thresholding can be used without size thresholding, or a low intensity threshold can be used with further thresholding on the size. The values l_d^*, and corresponding TPR*, S_T, and S_F, for which FPR* = 0.01, are shown in Fig. 2.8 and Table 2.5.

2.4.1.6 H-Dome Based Detection

The method based on the h-dome transformation (further referred as HD) was evaluated depending on the dome height h. The parameters of the method (see Section 2.3.2.5) were fixed to $\sigma_L = 2.5$, $\sigma_M = 6$, $s = 6$, and $N = 5000$, which maximize the TPR for the Type C image data at FPR* = 0.01. The results of the experiments are shown in Fig. 2.9. As described, the method estimates the object position and the variance of that estimation using a sampling procedure, bypassing the explicit creation of the map $\mathcal{C}$ [146]. The values h^* and corresponding TPR*, S_T, and S_F, for which FPR* = 0.01, are shown in Fig. 2.9 and Table 2.6.

2.4.1.7 Image Features Based Detection

This scheme (further abbreviated as IFD) creates the classification map $\mathcal{C}$ during Step 2 by combining the image intensities with local curvature information (see Section 2.3.2.6). Two types of the map $\mathcal{C}$ were considered in the experiments (with the resulting methods abbreviated as IFD_1 and IFD_2 respectively). In the first case, $\mathcal{C}$

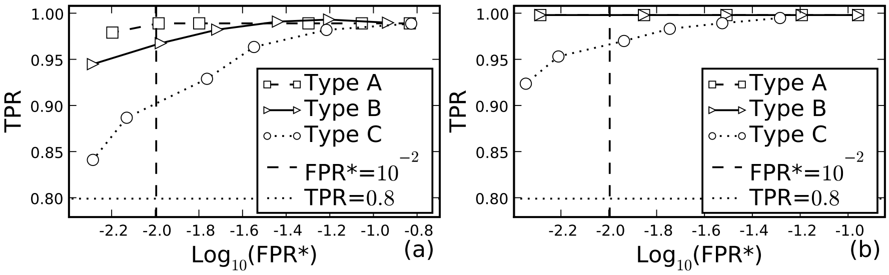

Figure 2.9. FROC curves for the HD detector in the case of round (a) and elongated (b) objects, depending on the values of the dome height h for different types of image data, at SNR = 2, and with parameters $\sigma_L = 2.5$, $\sigma_M = 6$, $s = 6$, and $N = 5000$.

Table 2.6. Optimal parameters and performance for the HD detector at SNR = 2 for parameters $\sigma_L = 2.5$, $\sigma_M = 6$, $s = 6$, and $N = 5000$.

Image	Round Objects				Elongated Objects			
Type	h^*	TPR*	S_T	S_F	h^*	TPR*	S_T	S_F
A	1.6	.99	.11	.05	1.4	.99	.01	.09
B	1.6	.97	.22	.05	1.4	.99	.01	.09
C	1.6	.90	.21	.05	1.2	.97	.16	.05

is given by the determinant of the Hessian matrix, $\det \mathbf{H}$, calculated at each pixel, with smoothing scale σ [159]. The second type of classification map $\mathcal{C}$ is obtained by pixel-wise multiplication of the values $\det \mathbf{H}(i,j)$ with the intensity values $J(i,j)$ (2.2). In the experiments, we used $\sigma = 2$, and the results are shown in Fig. 2.10 and Table 2.7.

2.4.1.8 AdaBoost

In order to test the performance of the ML approaches, starting with AdaBoost (abbreviated as AB) for the detection of round objects, we constructed a pool of 962 Haar-like features (see Section 2.3.3.1) using a 10×10 pixel subwindow, which was previously reported as optimal for similar applications [73]. Experiments with other subwindow sizes in the range of 8-12 pixels showed no significant difference in performance. For the detection of elongated objects, the subwindow size was fixed to 13×13 pixels, which consequently gives 2366 features. Even though the characteristic size of the elongated objects is doubled (compared to the round objects), the use of larger subwindow sizes, for example 21×21 pixels, degraded the AdaBoost performance. With the high spot density, the larger subwindows included the neighboring objects (equally frequently in the positive and negative training sets) and caused the problem with defining a clear decision boundary for these ML approach.

For the training stage, separate sets of synthetic images were created, and 4096 positive and 4096 negative samples (10×10 pixels) were extracted from each image type (A, B and C) containing round objects. The same training procedure was re-

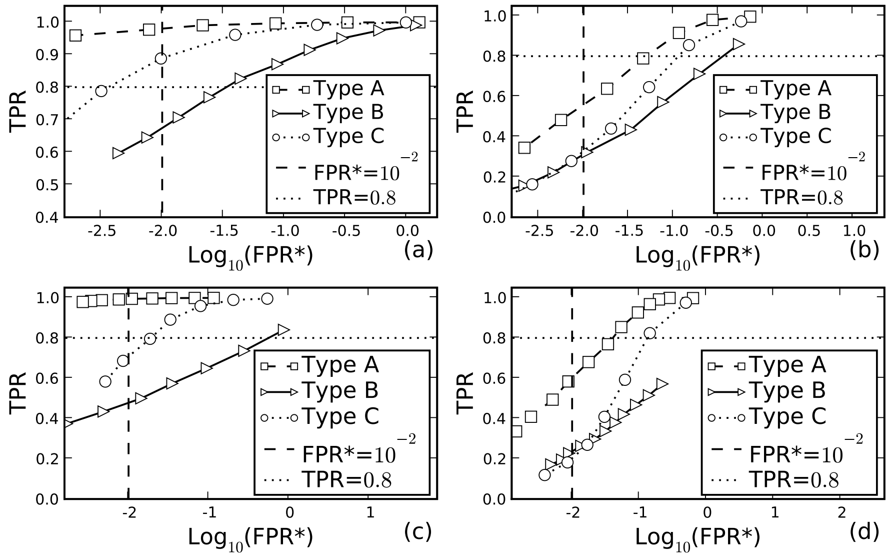

Figure 2.10. FROC curves for the IFD$_1$ detector in the case of the round (a) and elongated (b) objects, depending on the values of the threshold l_d and the type of image data, at SNR = 2, and for smoothing scale $\sigma = 2$. The same curves for IFD$_2$ in the case of round (c) and elongated (d) objects.

Table 2.7. Optimal parameters and performance for the IFD detectors at SNR = 2 and for smoothing scale $\sigma = 2$.

Image Type	Round Objects				Elongated Objects			
	l_d^*	TPR*	S_T	S_F	l_d^*	TPR*	S_T	S_F
	IFD$_1$							
A	.12	.98	0.67	.68	.21	.53	5.17	.42
B	.58	.67	1.23	.12	.71	.31	1.02	.06
C	.18	.89	2.51	.16	.28	.31	3.21	.26
	IFD$_2$							
A	1.33	.99	.03	.03	3.06	.59	.32	.03
B	33.34	.46	.01	.00	43.36	.23	.01	.00
C	1.95	.71	.36	.03	6.33	.19	.08	.01

peated for elongated objects. Four types of training were performed: using the samples from each image type separately, and using the combined training dataset, where 4095 samples were selected (in total) from type A, B and C images in equal proportions. The training was based on SNR = 2 (the worst case considered in this chapter). Training using higher-SNR images resulted in worse performance on lower-SNR images, as the number of features selected by AdaBoost became too small. Each trained classifier was applied separately to the synthetically created test images of all three types, with SNR in the range 2–4, and the classification results (sensitivity (TPR)

Table 2.8. Sensitivity and specificity of AdaBoost classification.

	Image Type A		Image Type B		Image Type C	
SNR	TPR	Spec.	TPR	Spec.	TPR	Spec.
	Trained using type A data (SNR = 2)					
2	0.994	0.995	0.999	0.930	0.965	0.987
3	1.0	0.996	1.0	0.922	1.0	0.989
4	1.0	0.995	1.0	0.919	1.0	0.992
	Trained using type B data (SNR = 2)					
2	0.914	1.0	0.991	0.977	0.690	1.0
3	1.0	0.999	1.0	0.977	0.998	0.999
4	1.0	0.999	1.0	0.977	1.0	0.999
	Trained using type C data (SNR = 2)					
2	0.996	0.992	0.999	0.902	0.999	0.979
3	1.0	0.990	1.0	0.910	1.0	0.982
4	1.0	0.991	1.0	0.901	1.0	0.982
	Trained using type A, B, C data combined (SNR = 2)					
2	0.988	0.998	0.998	0.942	0.962	0.994
3	1.0	0.997	1.0	0.939	1.0	0.995
4	1.0	0.998	1.0	0.940	1.0	0.993

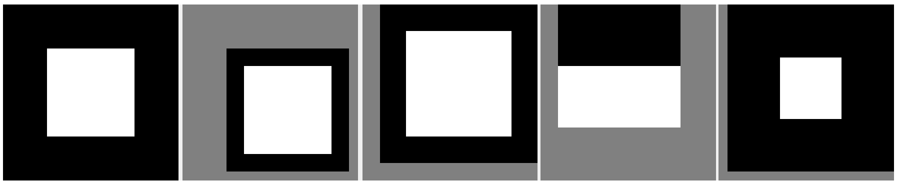

Figure 2.11. Example of the top-five features that were selected by AdaBoost in the case of the Type A training data.

and specificity) for 4096 positive and 4096 negative patches, extracted from these test images, are given in Table 2.8. In the experiments, the number of AdaBoost runs, N_{AB}, which corresponds to the number of features selected and used by the classifier, was fixed to 5. The top-five features selected during the training are shown in Fig. 2.11.

The behavior of the sensitivity and specificity was also investigated depending on the number of Haar-like features, N_{AB}, that are used for the classification. For this analysis, combined training (using the data of type A, B, and C) was performed, and the classifier was separately applied to the test data of each type. The results for different values of N_{AB} are shown in Table 2.9, where the last three rows also show the performance of the classifier trained using a reduced training set of 1002 combined samples (334 of each type).

In all these performance evaluation experiments, the classifier was applied to

Table 2.9. Sensitivity and specificity of AdaBoost classification depending on the number of runs.

	Image Type A		Image Type B		Image Type C	
SNR	TPR	Spec.	TPR	Spec.	TPR	Spec.
	$N_{AB} = 5$					
2	0.988	0.998	0.998	0.942	0.962	0.994
3	1.0	0.997	1.0	0.939	1.0	0.995
4	1.0	0.998	1.0	0.940	1.0	0.993
	$N_{AB} = 10$					
2	0.991	0.998	0.999	0.946	0.965	0.994
3	1.0	0.998	1.0	0.944	1.0	0.996
4	1.0	0.998	1.0	0.944	1.0	0.993
	$N_{AB} = 20$					
2	0.991	0.999	0.999	0.953	0.965	0.994
3	1.0	0.998	1.0	0.957	1.0	0.996
4	1.0	0.998	1.0	0.954	1.0	0.996
	$N_{AB} = 5$ and 1002 training samples					
2	0.991	0.999	0.999	0.953	0.965	0.994
3	1.0	0.998	1.0	0.957	1.0	0.996
4	1.0	0.998	1.0	0.954	1.0	0.996

image patches extracted from the positive and negative test images. In order to evaluate the performance of actual *detection* using this machine learning approach, we applied the classifier to each pixel in the images (based on a window of size 10×10-pixels around the pixel). The resulting classification map is a new image of the same size as the original, with each pixel being either "1" (if the corresponding image pixel was classified as belonging to an object) or "0" (if the pixel was classified as background). Before labeling the connected components and extracting the number of detected objects and their positions, the map was median-filtered with a round mask of radius 2 pixels in order to suppress too small clusters, and then a closing operation was applied with the 3×3 structuring element to fill small holes. The FROC curves for this detection procedure depending on the size threshold v_d of the clusters in the binary classification map C_B in the case of round and elongated objects are shown in Fig. 2.12. The behavior of TPR and FPR* depending on the number of features, N_{AB}, used in the detection is shown in Table 2.10. The parameters of the detection were optimized in order to get FPR$^* = 0.01$ when $N_{AB} = 50$. After that, the number of features N_{AB} was reduced (see Table 2.10) and the behavior of the performance measures studied. The optimal parameter values for the size threshold v_d are shown in Table 2.11.

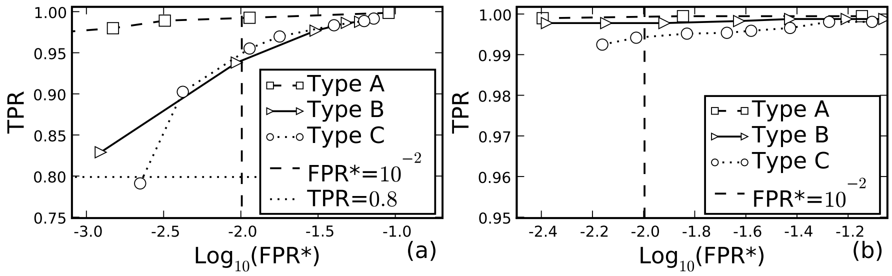

Figure 2.12. FROC curves for the AdaBoost detector in the case of the round (a) and elongated (b) objects, depending on the value of the size threshold v_d, at SNR = 2, and with $N_{AB} = 50$.

Table 2.10. Detection performance of AdaBoost depending on the number of selected features, N_{AB}, with training based on the combined image data (type A, B, and C) at SNR = 2.

	Image Type A		Image Type B		Image Type C	
N_{AB}	TPR	FPR*	TPR	FPR*	TPR	FPR*
5	0.995	0.013	0.912	0.037	0.806	0.019
10	0.996	0.014	0.929	0.041	0.818	0.022
20	0.994	0.013	0.921	0.022	0.789	0.019
50	0.994	0.011	0.926	0.016	0.810	0.018

Table 2.11. Optimal size thresholding parameters and corresponding performance for AdaBoost at SNR = 2.

Image	Round Objects				Elongated Objects			
Type	v_d^*	TPR*	S_T	S_F	v_d^*	TPR*	S_T	S_F
A	3	.99	10^{-3}	10^{-3}	2	.99	10^{-5}	.10
B	31	.94	.01	10^{-3}	18	.99	10^{-5}	10^{-3}
C	30	.94	.01	10^{-3}	12	.99	10^{-5}	10^{-3}

2.4.1.9 Fisher Discriminant Analysis

The classifier in this case (abbreviated as FDA) was trained using the same training data as in the case of AdaBoost. Using the labeled 10 × 10 image patches (for the round objects) and 13 × 13 patches (for the elongated objects), the kernels $\mathbf{w}$ for both types of objects were obtained (see Fig. 2.17(d,e)). Then, the sliding subwindow was used in order to classify every pixel in the image $\mathcal{I}$. The method produces the binary classification map $\mathcal{C}_B$ directly, so the performance of the detector was studied depending on the threshold v_d (which defines the size of the clusters of connected pixels in $\mathcal{C}_B$), and not the signal threshold l_d. The results are shown in Fig. 2.13 and the optimal parameter values are presented in Table 2.12. The size threshold,

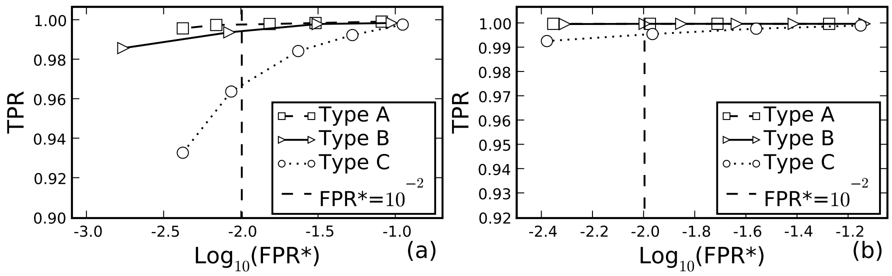

Figure 2.13. FROC curves for the FDA detector in the case of the round (a) and elongated (b) objects, depending on the values of the size threshold v_d and the type of image data, at SNR = 2.

Table 2.12. Optimal size thresholding parameters and corresponding performance for the FDA detector at SNR = 2.

Image	Round Objects				Elongated Objects			
Type	v_d^*	TPR*	S_T	S_F	v_d^*	TPR*	S_T	S_F
A	4.6	.99	10^{-5}	.01	3.0	.99	10^{-5}	10^{-2}
B	8.8	.99	10^{-3}	.01	5.6	.99	10^{-5}	10^{-2}
C	9.8	.96	10^{-2}	.01	12.4	.99	10^{-5}	10^{-3}

which in principle is an integer number (the minimum number of pixels a cluster in $\mathcal{C}_B$ should have to be considered an object), is real-valued in Table 2.12, due to the interpolation in order to obtain the value v_d^* for which FPR* = 0.01.

2.4.1.10 Comparison of All Detectors

The performance of all the described detectors was compared at the level of FPR* = 0.01 for the different image data at SNR = 2. The results are shown in Fig. 2.14. From the sensitivity analyses (see Tables 2.2-2.7, 2.11, 2.12), which was based on the comparison of ΔTPR and ΔFPR around the optimal signal thresholds for different detectors and data types revealed that the FDA and AB are superior to all other detectors and show the highest TPR* and the lowest sensitivity for all image data (Type A, B and C, SNR = 2). The WMP demonstrated the worst performance and additionally showed high sensitivity to parameter changes, together with the TH detector, which demonstrated high performance only for Type A and B data. The IFDs are quite sensitive to parameter changes and do not have sufficiently high TPR in the case of the elongated objects. The HD, SEF and MTH demonstrate high TPR* and low parameter sensitivity, but none of these three detectors is better than the other two for *all* types of data. Finally we observed that the difference in performance between the methods decreases when the SNR of the image data increases, and we found that for SNR > 5 all methods perform equally well (TPR = 1).

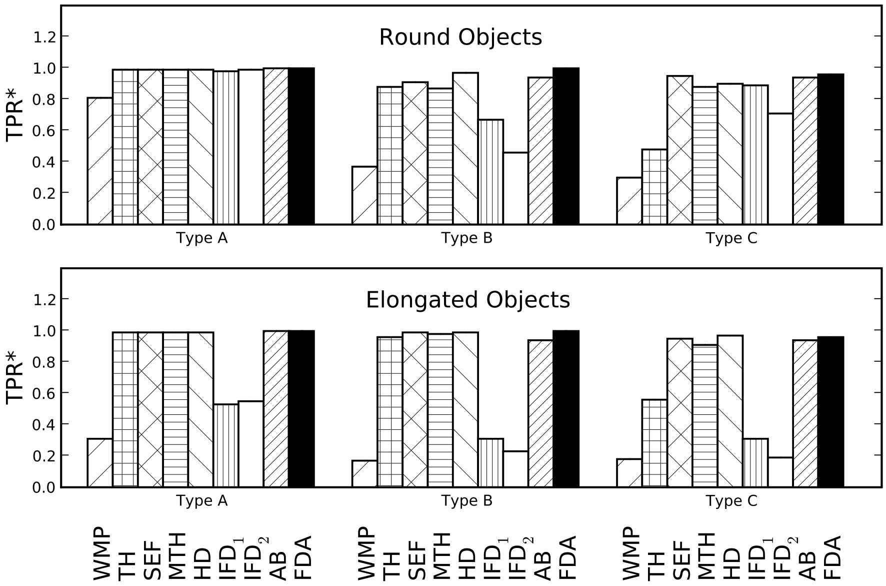

Figure 2.14. Maximum detection probabilities (TPR*) at the level FPR* = 0.01 for all the detectors applied to all three types of synthetic image data at SNR = 2 in the case of the round (top) and elongated (bottom) objects.

2.4.2 Evaluation on Real Image Data

2.4.2.1 Image Data

The described detection methods were also tested on real time-lapse fluorescence microscopy image data from several biological studies. The main goal of these studies was to estimate important kinematic parameters of subcellular particles in eukaryotic cells. To understand the molecular mechanisms underlying particle motility and distribution, it is essential to characterize in detail different dynamic properties, such as velocities, run lengths, and frequencies of pausing and switching of cytoskeletal tracks. This requires accurate tracking of individual particles, for which a wide variety of automatic tracking algorithms can be found in the recent literature [10, 17, 38, 52, 72, 74, 75, 128, 132, 141, 142]. In turn, these algorithms generally depend heavily on the performance of the spot detection stage, which forms an integral part of any tracking algorithm (see Section 2.1).

Two types of representative image data sets were selected for these experiments. The first showed moving microtubule (MT) plus-ends, which have a round or elongated appearance. MTs are hollow tubes (diameter of 25 nm) assembled from α/β-tubulin heterodimers, which frequently switch between growth and shrinkage [80,155]. The MT network is highly regulated and is essential to many cellular processes. In the experiments, growing ends of MTs were tagged with so-called plus-end-tracking proteins (+TIP), resulting in typical fluorescent "comet-like" dashes in the image se-

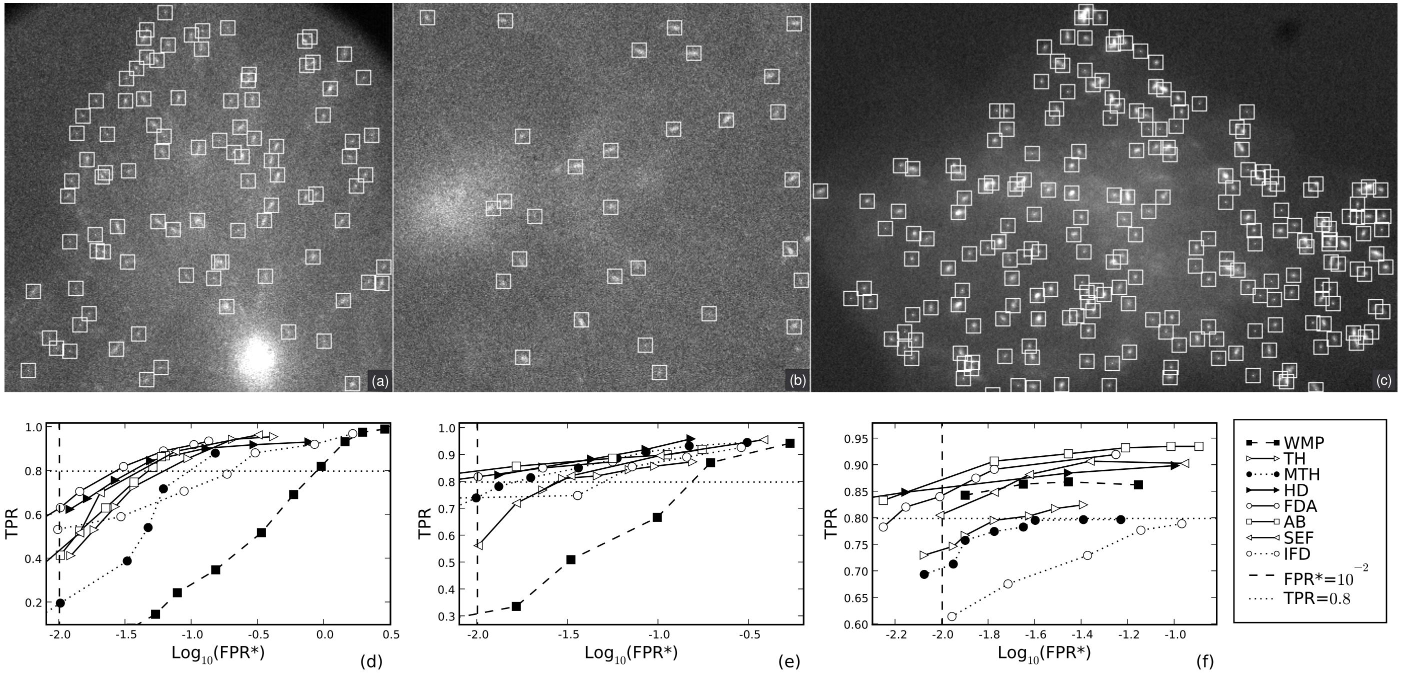

Figure 2.15. Examples of real fluorescence microscopy images (a, b, and c, compare Fig. 2.1) with manual spot annotation (white squares) by an expert biologist serving as ground truth. The corresponding FROCs (d, e, and f) of all the detection methods (with dependence on the same free parameters as in the experiments on the synthetic image data) are shown below the images (a, b, and c). In these plots, IFD represents the IFD$_1$ detector, which in the experiments on synthetic image data performed either similar to or better than the IFD$_2$ detector (see Fig. 2.14).

quences. In our study, COS-1 cells were cultured and transfected with GFP-tagged proteins [155]. A Zeiss LSM-510 confocal laser scanning microscope was used to acquire images of GFP+TIP movements at a rate of 1 frame per 1 or 2 seconds. The image sequences consisted of 30–50 frames of 512×512 pixels of size 75×75 nm^2 (see Fig. 2.15(a,b)).

The second type of image data showed a variety of GFP-labeled vesicles (Rab6 and peroxisomes), which have a round shape in the images. In this case, HeLa cells and PEX3-GFP fusion were used [58]. The HeLa cell line is the oldest cell line and is widely used for many different studies. Many variants of the HeLa cell line exist, including HeLa-R, with a so-called "round" phenotype, and HeLa-L, with a "long" phenotype. HeLa-L cells were used to study the dynamic properties of vesicles, and HeLa-R cells to study microtubule dynamics, microtubule and cell cortex crosstalk, and exocytosis [58]. Images were acquired on a Zeiss Axiovert 200M inverted microscope at a rate of 0.83 frames per second. The image sequences consisted of 100 frames of 1344×1024 pixels of size 64×64 nm^2 (see Fig. 2.15(c)).

2.4.2.2 Experiments and Results

For the experiments on real image data, the parameters of each detection method (except the thresholds l_d and v_d) were fixed to the same values as in the case of the experiments on synthetic data. Since the ground truth was not available for the real data, the results of the detection were analyzed by expert visual inspection and in comparison with manual analysis using MTrackJ [94].

The FROC plots for all the detection methods applied to two illustrative image data sets showing MTs (each image containing $\approx$ 80–100 spots at SNR $\approx$ 2–4) and one data set showing vesicles (containing $\approx$ 250 spots at SNR $\approx$ 3–8) are shown in Fig. 2.15. For the latter data set, all detection methods performed reasonably well, including the WMP detector, which performed notably worse on the MT data. In all cases, the two ML detectors (FDA and AB) and the HD detector showed the best overall performance. For visual comparison, the kernels obtained by FDA for the three mentioned real image data sets, as well as for the two types of synthetic data sets are shown in Fig. 2.17, where, for example, Fig 2.17(c) depicts the fact that the vesicle appearance in our images (see Fig. 2.15(c)) is more diverse compared to the microtubule data (Fig. 2.15(a, b)).

As an example, the results of all methods applied to an MT data set with SNR $\approx$ 2 are shown in Fig. 2.16. Manual annotation was extremely laborious and tedious in this case: visual comparison of several neighboring time-frames in the image sequence was necessary in order to establish object presence. Based on visual inspection of the results, it was found that the HD detector yielded the largest number of TPs and the smallest number of FPs. Here, in order to test the robustness of the ML approaches, the training was done using positive and negative samples obtained from another dataset (see Fig. 2.1(b)) with SNR $\approx$ 2–3. The results of this experiment imply that FDA is more sensitive to the training data: if the training is done using image data with different imaging conditions (SNR), the performance of the classifier can degrade. The AdaBoost algorithm, on the other hand, is less sensitive.

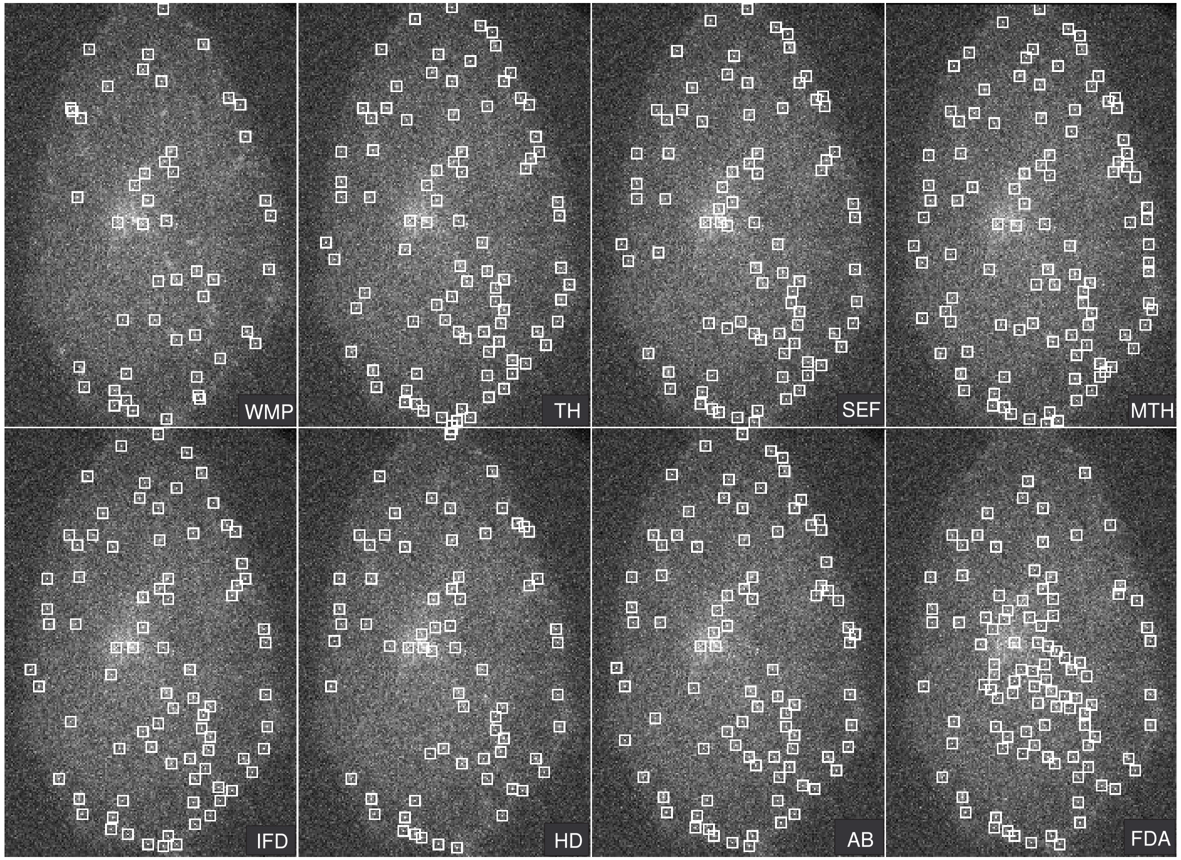

Figure 2.16. Results of applying all the described detection methods to real fluorescence microscopy image data showing GFP+TIP-labeled MTs at SNR ≈ 2. The HD detector yielded the largest number of TPs and the smallest number of FPs. Similar to Fig. 2.15, IFD represents the IFD_1 detector, which performed either similar to or better than the IFD_2 detector.

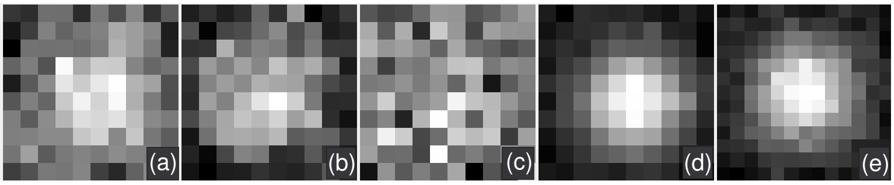

Figure 2.17. The FDA kernels for the MT data (a and b), vesicles (c), and the round and elongated objects from the synthetic data (d and e).

2.5 Discussion and Conclusions

In this chapter we have evaluated the performance of six unsupervised and two supervised detection methods that are frequently used in practice for the detection of small spots in fluorescence microscopy images. It was shown that all of the described methods follow a "three-step" signal processing procedure, but implement each of these steps in a specific way. In order to build an accurate and robust detector for a particular application, a careful selection of the algorithms for each of the steps is necessary. The results from experiments on synthetic images as well as real image data from two biological studies indicated that no detector outperforms all others in all considered situations. Overall, the supervised (machine learning) methods performed better on the synthetic images as well as on the real image data, but the differences in the performance were not large compared to some of the unsupervised methods.

In order to study the influence of small changes in the parameter settings of the detection methods, a sensitivity analysis was carried out by computing the resulting rate of change in TPR (the true-positive ratio) and FPR (the false-positive ratio) around the empirically determined optimal signal threshold, for two types of objects (round and elongated). From the experiments on the synthetic images at very low SNR (≈ 2), we found that the AB (AdaBoost) and the FDA (Fisher discriminant analysis) detectors are superior to all other detectors, in that they show the highest TPR (at very low FPR) and the lowest sensitivity to parameter changes, for all types of image data considered: uniform background (Type A), background gradient (Type B), and cluttered background structures (Type C). Of all the unsupervised detectors, the WMP (wavelet multiscale product) detector showed the worst overall performance and, additionally, high sensitivity to parameter changes. Similarly, the TH (top-hat based) detector showed high performance only for Type A and Type B data. The HD (h-dome), MTH (morphological top-hat), and SEF (spot-enhancing filter) based detectors showed high TPR and low parameter sensitivity, but none of them was better than the other two for all data types. Both variants of IFD (the image-feature based detector) were quite sensitive to parameter changes and did not show high TPR in the detection of elongated objects. Finally, we also observed from these experiments that for SNR > 3, the difference in performance of all the detectors rapidly decreases.

From the experiments on real fluorescence microscopy image data, it was confirmed that the actual performance of the detection methods depends on the application. For the microtubule data, which contained round or elongated objects of almost identical sizes, we arrived at the same conclusions as in the case of the synthetic image data. For the vesicle data, however, the ranking of the detectors was found to be slightly different. These images have a higher SNR (≈ 3–8) but contain spots of varying sizes. In this case, the detection methods that have parameters that explicitly relate to spot size, such as the TH and MTH detectors, showed quite poor performance. Once their parameters are set, these detectors expect spots to be of similar size. Similarly, the image-feature based IFD detector works well only when all the spots have very similar appearance in terms of the features considered. On the other hand, detectors such as SEF and HD do not model the spots exactly, and because of that allow some more variation in the appearance of spots. Moreover, the WMP detector, which also does not assume any specific object shape, demonstrated much better performance for such datasets.

Based on our extensive experiments, we conclude that when a detector with overall good performance is needed, the supervised AB or FDA detectors or the unsupervised HD detector are to be preferred. The main disadvantage of the supervised methods is that they require a training stage, which involves the extraction of positive and negative samples beforehand. As was shown, the training should not be done using only clearly visible spots in image regions with high local SNRs. On the contrary, in order to achieve good classification performance, it must also include a lot of hardly visible objects. Such manual annotation is extremely tedious, time consuming, and observer dependent. Spots may be more or less identical within one data set, but may differ in appearance from one data set to another, due to the different experimental and imaging conditions. Because of that, one would have to repeat the training (or correct it) when new data sets arrive. The preparation of training samples requires manual annotation of thousands of objects in order to achieve sufficient discriminating power, which itself is a manual detection that biologists would be happy to use, without considering further automated analysis. Taking this into account, the unsupervised HD detector is much easier to use in practice. Finally, when the SNR is sufficiently high (> 5 as a rule of thumb), the other unsupervised detectors perform just as well, and require only minimal adjustment of their parameters to the specific application.