Chapter 3.2

AUTOMATED ANALYSIS OF KYMO-GRAPHS WITH KYMOGRAPHCLEAR AND KYMOGRAPHDIRECT
3.2.1 Abstract

A kymograph is a single image that represents the motion of a dynamical process originally captured as a time series of images. Kymographs are obtained by plotting the intensity along a track as a function of time and are widely used in live-cell imaging. Extracting quantitative information from kymographs has proven to be time consuming and difficult to automate. In this chapter, two complementary tools to generate kymographs and to automatically analyze them are presented. KymographClear is a macro toolset for ImageJ to generate high-quality kymographs and allows automatic color coding of particles depending on their direction of movement. KymographDirect is a stand-alone application to extract quantitative information from a kymograph, in an automated way, with unprecedented accuracy and reliability. These tools are used in Chapter 4.1.
INTRODUCTION

Directed motion is ubiquitous in living cells and governs crucial life processes such as DNA synthesis, intracellular transport, cytoskeletal rearrangements, and mitosis (1). Using fluorescence microscopy, or other types of optical microscopy, it is possible to capture the dynamics of these processes, by generating time series of 2- or 3-dimensional images. For a proper understanding, it is crucial to rigorously quantify the dynamics (e.g. determine directions, speeds and fluxes), which can be a difficult and time-consuming task. Single-particle tracking (2) provides the most detailed information, but is severely hampered by high particle density and low signal-to-noise ratios (SNR): in many cases, image quality is not sufficient to allow accurate particle tracking. In such cases, kymographs are a convenient means to obtain insight in motility. A kymograph is a time-space plot, in which the intensity along a given track is plotted as a function of time (Fig. 1A, B). Particles moving with constant velocity show up as straight lines, with a slope directly related to velocity and intensity dependent on the number of particles moving together. The density of lines is a measure of particle flux. Kymographs thus provide direct, qualitative insight in key motility parameters. In principle, quantitative information can be extracted (3), but effective software tools for the automated analysis of kymographs are lacking. Consequently, kymographs are generally analyzed by human, visual inspection, which is time consuming and prone to user-bias, potentially resulting in irreproducibility. The algorithms that have been proposed so far only work for kymographs with high SNR and low particle densities (4, 5). In this chapter, two freely available software tools are described - KymographClear and KymographDirect - the former to generate kymographs and the latter to perform automated, quantitative analysis of kymographs even in conditions with low SNR and high particle density.

METHODS

3.2.2 KymographClear

KymographClear is a macro toolset for the popular, open-source image-processing tool ImageJ (6) that enables the generation of high-quality kymographs using sub-pixel interpolation on any track geometry, including curved tracks, a feature lacking in other ImageJ kymograph tools (Fig. 1A, B). KymographClear generates reliable kymographs with accurate intensities that can be used for further
quantitative analysis using the accompanying software tool KymographDirect. Kymographs can be corrected for background signals by evaluating the background signals locally. KymographClear makes use of a Fourier-filtering algorithm to automatically color code particles depending on their direction of movement: moving forward, backward, or remaining static (Fig 1C). This feature facilitates visualization of distinct dynamics and thus helps interpretation of the underlying cellular processes.

Figure 1: Analysis of kymographs with KymographClear and KymographDirect. KymographClear: (A) A track (red dashed line) is drawn on a time-lapse sequence of images, from which a kymograph is generated. (B) The time-lapse image sequence used here is a series of wide-field fluorescence images obtained from the phasmid cilia of a living *C. elegans* nematode. The animals expressed GFP-containing variants of the kinesin motor OSM-3. (C) Fourier filtering allows the separation of forward (red), backward (green) and static motion components (blue) in the kymograph. KymographDirect: (D) Identification of individual forward and backward trajectories. (E) KymographDirect provides 19 different outputs, including particle coordinates, time and spatial dependence of particle velocities and intensities. Presented here are the local particle-averaged velocity (black) and average intensity (red) of forward-moving particles in kymograph B (solid lines represent average and dashed lines standard deviation). Horizontal scale bars (in A-D) correspond to 2 µm, the vertical ones to 4 s.
3.2.3 KymographDirect

A stand-alone application, KymographDirect, was developed to analyze the kymographs generated with KymographClear. KymographDirect is able to automatically detect trajectories of moving particles in kymographs (Fig. 1D). The program uses a similar Fourier-filtering approach to separate different motion components, which at the same time increases the SNR by a factor of two, further improving reliability and accuracy of trajectory tracking. The performance of the application is robust even for low-SNR kymographs reflecting frequently crossing particles moving with changing velocities (Fig. S10-S13). KymographDirect uses the extracted trajectories for further quantification and generates 19 different outputs important for the interpretation of motility, including trajectory coordinates, particle velocities and intensities as a function of time or position, and statistical analyses of these quantities (Fig. 1D, E and Fig. S7-S9); these outputs can be saved for further analysis using other data-processing tools. Special care has been taken to correctly evaluate the intensities of moving particles, by correcting for signals due to background as well as static particles.

The application has been validated to perform reliably under conditions mimicking imaging conditions often found in literature (Fig. S10-S13). Trajectories are typically extracted with sub-pixel accuracy even for SNR close to 1 (Fig. S10B, S11B, S13C). Even for low-SNR kymographs, the application is capable of extracting the majority of trajectories. At SNRs above 2, at least 80 to 90 percent of trajectories are reliably uncovered (Fig. S11D, S13D). The algorithm’s ability to precisely determine velocities depends on the stochasticity of the process under study; in most cases the uncertainty encountered is at most a few percent (Fig. S10C, S11C).
SUPPORTING INFORMATION

3.2.4 Introduction

KymographClear and KymographDirect are two toolsets for the analysis of directed motion in image sequences using kymographs. Both tools are controlled by a graphical user interface. KymographClear and KymographDirect are designed to be used in a sequential way as schematically depicted in Figure S1.

KymographClear is an ImageJ (3) macro toolset to generate accurate kymographs from an image sequence, allowing motion to be tracked along tracks with any geometry. The output of this tool is a kymograph, an array of image intensity as a function of time and location along a track chosen by the user. KymographClear produces kymographs as text and image files. In addition, it allows Fourier-filtering of the kymograph such that motion in different directions can be color coded (Fig. S1). KymographClear contains a function that reports the average background intensity in the images (outside the kymograph track), useful for background subtraction in fluorescence images. KymographClear has been developed with ImageJ 1.48k; the source code is available at www.nat.vu.nl/~erwinp/downloads.html.

KymographDirect is a stand-alone application for the automated analysis of the kymographs generated by KymographClear. It allows for tracking the position of individual objects in forward and backward directions, quantification of their intensities, velocities and statistical analysis of these quantities. In case fluorescence imaging is used to generate the data, bleaching and background corrections can be applied. KymographDirect has been written in National Instruments LabVIEW 12.0f3 32-bit. The stand-alone executable (for Windows and Mac OS) and the source code are freely available at www.nat.vu.nl/~erwinp/downloads.html. KymographDirect is licensed under the GNU general public license version 3.
GENERATING KYMOGRAPHS WITH KYMOGRAPHCLEAR

3.2.5 The kymograph-generation algorithm

Although ImageJ plugins for kymograph generation have been available, all plugins tested do not make use of sub-pixel capabilities, a recently added feature in ImageJ (from version 1.46a, November 2011). An important consequence is that kymographs generated by these plugins do not always report the correct length for in particular curved tracks, resulting in substantial error in velocity determination. A new ImageJ macro toolset, KymographClear (that is described in this chapter) allows for the accurate generation of kymographs, along tracks with any geometry, including curved lines, making use of the sub-pixel capabilities of the newer version of ImageJ. KymographClear consists of several macros accessible from the graphical user interface of ImageJ. How to install KymographClear is explained in the user manual provided with the installation files.

The two macros “open a sequence and compute average image” and “open a sequence and compute max intensity image” open an image sequence from a file location designated by the user and computes an averaged or maximum-intensity image (Fig. S1A).

On this averaged or maximum-intensity image, the user can define a track along which the kymograph is determined. The user can select a straight line, a segmented line, a freehand line or a spline. The following macro Kymograph generation uses the selected track and interpolates it; the interpolation is carried out by the ImageJ Interpolate selection function with single-pixel intervals. Subsequently, the user selects the line width \(lw\) of the track to analyze (an integer number of pixels). Then, the macro generates a kymograph from all images in the sequence by evaluating the intensity along the track, using the built-in getProfile function (Fig. S1B). The macro saves the kymograph data into a text file to avoid any unwanted scaling or digitization by ImageJ (Fig. S1C). In addition, an image file of the kymograph (in 32-bit TIFF format) is generated.
Automated analysis of kymographs with KymographClear and KymographDirect

KymographClear

1. Image sequence
2. Selection of an area to evaluate the background
3. Averaged image or maximum intensity image and track selection by the user
4. Kymograph generation
5. Kymograph with forward-moving, backward-moving, and static components discriminated
6. Image and text file of the kymograph
7. File with linewidth setting
8. File with background evaluation

KymographDirect

1. Selection of the direction of motion to analyze and selection of the settings to use
2. Background and bleaching correction
3. Automated tracking of the position of single particles in the kymograph
4. Automated analysis
5. Result saved
3.2.6 Discrimination of forward, backward and static components by Fourier filtering

In many cases, image sequences of directed motion contain objects moving along a track in both forward and backward directions and those that remain static. Kymographs from such sequences can be difficult to interpret and analyze, because of overlapping and crossing features. KymographClear employs Fourier filtering to separate such different motion components in kymographs. In the Fourier transform of a kymograph, forward- and backward-moving components are localized in different quadrants, allowing straightforward separation of motion components (Fig. S2A-B). Images reflecting only forward motion or backward motion can be obtained by masking the Fourier-transformed kymographs (setting the pixel values of two diagonally opposed quadrants to zero), followed by inverse Fourier transformation. An additional benefit is that noise is reduced by a factor of two in case two diagonally opposite quadrants are used as mask.

Static components can be filtered out in a similar way. In case the kymograph is oriented such that the time axis is vertical and the position axis horizontal, static objects result in vertical lines in the kymograph. In the Fourier-transformed kymograph, they are represented by a horizontal line corresponding to (temporal) frequency zero (Fig. S2B). Selecting only this line and inverse Fourier transformation allows for filtering the static component. In Figure S2C the effectiveness of kymograph Fourier filtering for discriminating static, forward and backward moving objects is demonstrated.
Figure S2: Effect of Fourier filtering on a kymograph. (A) Simulated kymograph displaying forward motion (lines traced from top left to bottom right), backward motion (lines traced from top right to bottom left) and static pools (vertical lines). (B) Absolute value of the Fourier transform of kymograph A. Highlighted in color are the regions containing information on forward-moving (red), backward-moving (green) and static (blue) components. (C) Kymograph with forward-moving, backward-moving and static components extracted using Fourier filtering and displayed in red, green and blue, respectively.

Fourier filtering can cause ringing artefacts at the edge of an image. This problem can be overcome by Fourier transforming an image paved with the kymograph as depicted in Figure S3, where each edge of the kymograph is matched with a corresponding copy properly oriented. The algorithm used in KymographClear applies a Fourier transform on this paved image, masks the Fourier-transformed image, applies an inverse Fourier transform and selects the kymograph in the center of the resulting paved image.

Results of the different filters to separate backward motion from forward motion and static particles are presented in Figure S2C. Although sharper masks can be applied to improve filtering efficiency (7), they are not used in this specific application, because these filters can suppress important characteristics of the dynamics, such as fast velocity changes.

Apart from the unfiltered kymographs, the Kymograph generation macro creates three image files of the forward- and backward-moving and static components extracted by Fourier filtering and a fourth, merged image displaying the three components in different color channels (Fig. S1D).
3.2.7 Background evaluation

For image sequences obtained with fluorescence microscopy, it can be desirable to correct the signal amplitude for background. Because of potential non-uniformity across the image, background correction needs to be performed carefully. In our program the background is evaluated in a user-selected region of interest, next to the selected track. Subsequently, the Background macro evaluates the average pixel value of the region of interest for each image in the sequence separately (Fig. S1E) and generates a file in which this information is stored (Fig. S1F).

**Figure S3: Reduction of “ringing effects”**. In order to reduce ringing effects caused by Fourier filtering, a Fourier transform is applied on an image paved with nine kymographs oriented as indicated. After filtering only the center kymograph (blue frame) is retrieved.
KYMOGRAPHDIRECT AUTOMATED KYMOGRAPH ANALYSIS

A major drawback to using kymographs is their analysis can be tedious, time consuming and user biased. Automated analysis algorithms have been published (5, 6), but they are limited in their application to images with high SNR, rarely obtained from cells and living animals. KymographDirect is a software tool that automates kymograph analysis for a wide range of applications and allows accurate determination of tracks even at very low SNR. It provides background and bleaching correction, automated line detection, velocity and intensity evaluation along detected lines as well as statistical analysis of these quantities. Large data sets can be analyzed in a short time and user bias is minimized.

3.2.8 Preliminary step for kymographs generated with fluorescence imaging: background and photobleaching corrections.

Before the kymographs can be analyzed, they need to be corrected for two effects that can together lead to distortion of fluorescence signals: (A) background and (B) photobleaching of the particles of interest.

(A) Background signals due to out-of-focus fluorescence light, autofluorescence of the sample, scattered excitation light or electronics, result in unwanted offset of the measured fluorescence intensity. To correct kymographs for background, the stored (section 3.2.7) average pixel intensity values of a region next to the track of interest are used. Typically, the background intensity decreases over time due to photobleaching, which affects some of the background sources. In our experience, the time evolution of the background signal can be well described by an exponential decay with offset. The background-correction algorithm performs a fit with such a function to the background signal and subtracts the fitted values from the kymograph.

(B) During fluorescence imaging, the continuous excitation of fluorophores results in photobleaching, gradually decreasing fluorescence intensity of the particles of interest. To correct background-corrected kymographs for photobleaching, the kymograph is divided by a correction function. This function is obtained from the kymograph by fitting the position-averaged intensity as function of time with an exponential decay. This correction is optional and should be applied only in situations where the number of fluorophores is constant.
3.2.9 Discrimination of forward- and backward-moving components by Fourier filtering

Discrimination of forward- and backward-moving components is performed in the same way as described in section 3.2.6 except that in KymographDirect background and photobleaching-corrected kymographs are used (when selected by the user). In later stages of the analysis, the program only considers the user-selected direction of motion.

3.2.10 Tracking individual particles

Linear motion of a particle results in a line in the corresponding kymograph, with slope directly related to the particle velocity. The tracking section of KymographDirect allows for the automated detection of lines caused by individual particles moving along the selected track. In brief, the tracking algorithm operates as follows: first, the particle average velocity is estimated for each position on the track; second, particles are detected and their trajectories in the kymograph are determined (Fig. S1G). The fidelity of tracking is substantially increased by making use of an estimate of the local particle-averaged velocity.

To estimate the average velocity at a given position, the intensity information between adjacent vertical pixel lines in the kymograph, corresponding to adjacent positions on the track, is correlated. Direct cross-correlation of the intensity between lines has been performed before (5), but often does not lead to robust results, because of image noise and signal variation. The algorithm follows the following sequence:

- Vertical kymograph lines (i.e. intensity as a function of time) are low-pass filtered to decrease noise (Fig. S4A).
- Peaks are detected with sub-pixel (time) resolution in these vertical kymograph lines, using an intensity threshold set by the user.
- The peaks in the vertical (time) kymograph lines are replaced by identical, normalized triangular functions centered at the detected peak position (Fig. S4BC).

(Continued on next page)
• Cross-correlation between two successive vertical pixel lines intensity signals obtained in the previous step: this gives an estimate of the average time particles need to translate the width of a pixel along the track, which can be converted to a local, particle-averaged velocity.

• Finally, to improve the robustness of the algorithm, the location-dependent velocity is low-pass filtered and fitted with a third order polynomial, which is the final output of this step (Fig. S4D).

**Figure S4: Evaluation of average local velocity in a kymograph.** (A) Example of an intensity versus time signal on a given position as obtained from Figure 1B kymograph (red) and the same signal processed by the first order low pass filter of the algorithm (black). (B) At each position of a peak found in the filtered signal, the algorithm places a triangular function of fixed height and width. (C) Example of two signals transformed into triangular functions obtained from adjacent pixel positions (black position n, red position n+1); the delay observed between these two signals signal is evaluated by autocorrelation. (D) Average velocity versus position of Figure 1B kymograph as obtained by the algorithm.

In case particles move very slowly, less than one pixel per ten frames, the cross-correlation algorithm as described above fails. Local velocity can then be obtained robustly by using the same algorithm on 90 degrees rotated kymographs. This version of the algorithm can be selected by the user, when low velocities are expected.
The algorithm to find individual trajectories consists of the following steps:

- Peaks are automatically detected in the vertical (position constant) lines of the kymograph obtained in section 3.2.3 with sub-pixel (time) resolution, using an intensity threshold set by the user. For low signal-to-noise ratio data, the kymographs can be low-pass filtered (along the time axis) prior to peak detection.
- The algorithm selects the highest intensity peak in the first vertical line and searches for a peak belonging to the same trajectory in the neighboring kymograph line, within a window set by the location-dependent velocity obtained in the previous step in the program (Fig. S5).

**Figure S5: Search for trajectory.** After the algorithm has found local intensity peaks in the kymograph (red dots), it links these peaks given the local average intensity; the search to link one peak to another is therefore limited to a window oriented toward the most probable velocity (blue triangle).

This part of the algorithm works as follows. Given a peak position $x_0$ at time, the peak in the neighboring line (at position $x_0 + 1$) is allowed to be located between time point $t_0 + 1/(2V)$ and $t_0 + 2/V$, with $V$ the average velocity of particles at position $x_0 + 1$. In case multiple peaks are found within this window, peaks are weighted with the function $(t-(t_0+1/V)+1) \exp(-(t-(t_0+1/V))-1)$, which is a skewed bell function with a maximum centered on the most probable peak, i.e. $t_0 + 1/V$. The peak with the highest weight is selected for further analysis.
In case no peak is found in the previous step, the algorithm looks for peaks in the next four vertical pixel lines, in the same way as the previous step.

In case a peak is found that is connected to the previous one, the algorithm looks for a connected peak in the subsequent kymograph position line and so on, constructing the trajectory point by point.

In case no further peak is found, the positions of the peaks found are gathered into a segmented line.

Then, the algorithm repeats this process, using peaks that have not been used so far, until there is no peak left.

For all the segmented lines found, the algorithm checks for potential overlap and proximity, erases those that are too similar and connects lines that follow upon each other.

Finally, the algorithm filters the segmented lines for a minimal length of four pixels along the position axis.

3.2.11 Determination of velocities

In a kymograph, the velocity of a moving particle can be readily deduced from the slope of the kymograph line belonging to the particle. In many applications of kymographs, only straight lines were fitted to kymograph lines, disregarding potential changes in velocity along the way. KymographDirect was explicitly designed to determine location-dependent velocities, requiring a more complex algorithm to accurately determine velocities. Our algorithm consists of the following steps:

In case a trajectory contains less than 30 points, additional points are added by linear interpolation, adding one point in the middle of each pair neighboring points. This is repeated until the number of points is larger than 30. To avoid artifacts due to edge effects when fitting trajectories, points are added to both beginning and ending of each trajectory. These extra points are only used for fitting. For the beginning of a trajectory, the points are obtained by reflecting points 2 to 10 with respect to an origin of point reflection. This origin of point reflection is the time value obtained at the position of the first point from a linear fit to points 1 to 10 (Fig. S6). The extra points at the end of the trajectory are obtained in the equivalent way. As a result, each trajectory is added with 18 points.
Each trajectory is fitted over a sliding window of 19 points with a third-order polynomial. The local slope of the trajectory is determined by calculating the slope at the tenth (middle) point of each fitted polynomial. The location-dependent velocity of the trajectory is obtained by sliding the 19-point window from one position to the next over the whole trajectory.

- The location-dependent velocity is converted to the proper units and scaling, and low-pass filtered to reduce noise.

3.2.12 Determination of particle intensities

Another key property of particle trajectories is their location- or time-dependent intensity. In case the particles are only moving in one direction, a simple pixel-intensity summation around the particle position is sufficient to evaluate the fluorescence intensity. In the general case, however, different particles move in opposing directions, or are static. In this case, summation is not sufficient since it does not separate the signals caused by these different contributions. Use of the Fourier-filtered kymographs can be beneficial, but care has to be taken that the pixel intensities in the filtered kymographs are correct. The different components of motion are located mostly in clearly separated regions of the Fourier transform (at non-zero spatial frequencies). However, the summed image intensity,
containing all motion components, shows up in the Fourier transform at zero spatial frequency. This value has to be taken into account in order to determine the absolute intensities of the different motion components (and not only the relative intensity). This problem is solved by assuming that kymographs contain time points at each position that do not contain intensity due to moving particles. At each position (in the inverse transformed image), the values are offset corrected by setting the average value of the 3% dimmest time points to zero.

After determining the correct intensity of the Fourier-filtered kymographs, the algorithm determines particles intensities by adding the intensities of several pixels along the position axis centered on the particle position. The number of pixels to be considered depends on the actual particle size, the resolution of the microscope and the particle velocity (due to blurring), all parameters that are specific for each experiment. I use a default value of five pixels, which can be altered by the user.

EXAMPLE OF A KYMOGRAPH ANALYZED WITH KYMOGRAPHDIRECT

Once the positions, velocities and intensities of the particles have been estimated, it is possible to analyze the data further. Spatial and temporal dependencies of these quantities can be computed, as well as the corresponding distributions. Figures S7, S8 and S9 show the different results KymographDirect can display for the kymograph used in Figure S1; Figure S7 display the trajectories detected in the Fourier-filtered kymographs, Figure S8 focuses on the time dependence of particle velocities and intensities, whereas Figure S9 focuses on the spatial dependence of particle velocities and intensities and the time-averaged intensity along the track. This data has been generated from a fluorescence image sequence of *C. elegans* chemosensory cilia, on animals expressing a GFP fusion with OSM-3 kinesin, a motor moving along the chemosensory cilia; the lines in the kymograph correspond to groups of about tens of these kinesins moving together. The results displayed can be saved in separate data files, including the individual data points displayed in the histograms of Figure S8. The standard deviation, the standard error of the mean and the number of events are saved together with the data corresponding to the averages displayed Figure S9.
Figure S7: KymographDirect screenshots of the overlay of the kymograph Fourier filtered for forward (A) and backward (B) motion and the corresponding trajectories detected (blue lines). The trajectory in red is used to highlight its corresponding contribution in the result graphs (Fig. S8, S9).
Figure S8: Time dependence of particle velocities and intensities (KymographDirect screenshots). (A) Top: velocities versus time of particles moving in the forward direction. Bottom: histogram of time-averaged velocities of particles. (B) Same as A, for particles moving backward. (C) Top: intensities of individual particle moving forward versus time. Bottom: histogram of time-averaged intensities of particles. (D) Same as C, for particles moving backward.
Figure S9: Spatial dependence of particle velocities and intensities (KymographDirect screenshots). (A) Top: velocity versus position of individual particles moving in the forward direction. Bottom: local, particle-averaged velocity versus position (red line) and corresponding standard deviation (dashed blue line). (B) Same as A, for particles moving backward. (C) Top: intensities versus position of individual particles moving forward. Bottom: local, particle-averaged velocity versus position (red line) and corresponding standard deviation (dashed blue line). (D) Same as C, for particles moving backward. (E) Time-averaged intensity as a function of position on the track (red line) and corresponding standard deviation (dashed blue line).
VALIDATION OF THE KYMOGRAPHDIRECT TRAJECTORY DETECTION AND VELOCITY QUANTIFICATION ALGORITHMS

In order to test the validity of the kymograph-analysis tools, they are tested on a variety of simulated data, mimicking different experimental conditions, ranging from sparse particles to densely crowded ones, with varying signal-to-noise ratios, particle crossings and displaying more or less stochastic motion behavior.

3.2.13 Single-particle tracking – effect of stochasticity in motion

The tests will start with a straightforward simulation of a kymograph from an individual, moving particle. In this simulation the effects of stochastic variations in particle velocity and signal-to-noise ratio (SNR) are explored (Fig. S10A).

Figure S10: Validation of the kymograph-analysis tools on data mimicking single-particle trajectories. (A) Examples of kymograph generated with stochastic acceleration of spread $\alpha$ ranging from 0 to 0.25 pixel/(unit time) (7) and with various signal-to-noise ratio (SNR). (B) Average distance between detected trajectories and input (simulated) for different stochastic accelerations of spread $\alpha$ versus SNR. Note that even for high SNR and low stochasticity a residual distance remains. This is due to pixelation effects creating a small offset between the simulated trajectory and the one detected by the algorithm. (C) Average relative error of the software in measuring particle velocity for different stochastic accelerations of spread $\alpha$ versus SNR.
To test these effects a particle is tracked that is moving with an initial velocity of 2 pixels per frame, undergoing a stochastic, normally distributed acceleration, with a spread ranging from 0 to 0.25 pixel frame$^{-1}$ (7). In addition, particle and background intensities are Poisson-distributed (with SNR defined as the ratio between the average intensity of the particle and the standard deviation of the background). Example kymographs are shown in Figure S10A.

Figure S10B shows that the algorithm can track the particles with sub-pixel accuracy, even at SNRs below 1 and for the most stochastic trajectories (Fig. S10B). The ability of the algorithm to accurately find trajectories at such low SNR is substantially helped by Fourier filtering and low-pass filtering. Apart from reliably tracking locations, the algorithm also performs well in determining velocities (Fig. S10C), in particular at relatively constant velocities.

### 3.2.14 Simulation of cell crawling – performance under crowded conditions

Kymographs are often used for visualization of cells crawling over surfaces (8, 9, 10, 11). Quantification of the motility parameters has, however, rarely been performed, because of difficulties with data and image analysis. Image sequences of this process obtained with fluorescence microscopy, are generally very crowded, yielding little contrast, preventing straightforward tracking of individual particles. When differential interference contrast (DIC) microscopy is used, the many structures in the cell also result in kymographs overloaded with partly overlapping lines, which impedes analysis of the kymographs. Our automated kymograph analysis tools substantially improve this and allow reliable, quantitative analysis of kymographs of such processes. It provides the opportunity to follow many particles individually and therefore measure accurately how a cell deforms locally and understand what molecular mechanisms are driving this process.

Experiments of cell motility were simulated, typically characterized by a large number of partly overlapping, moving fluorophores or particles and a rather low stochasticity in the motion. An assumption was that a cell was crawling on a surface. In a first set of simulated data, fluorescence microscopy images of about 30 particles were mimicked, randomly distributed along a line of motion that is used to generate the kymographs. The fluorescence intensity of the particles is Poisson distributed, each particle being assigned a random time-averaged intensity ranging from 1 to 3-fold the average amplitude of the dimmest particle. To
test a wide range of SNRs, each kymograph was simulated with different relative magnitudes for both fluorescence intensity and background (Fig. S11A), in the same way as in the single-particle tracking case above. The results are similar to the previous case where only a single particle was followed. The algorithm reveals tracks reliably with sub-pixel resolution, even at low SNR (Fig. S11B) and velocities are determined accurately, within a few percent (Fig. S11B-C). The algorithm is also very efficient in finding trajectories: in this simulation, at SNRs above 3 more than 90% of the trajectories are revealed and precisely detected (Fig. S11D); even in the range of SNR = 1-3, the majority of trajectories is found. The analysis of these simulated kymographs shows that our algorithms are very efficient in obtaining quantitative parameters from crowded, low SNR kymographs.

**Figure S11: Validation of the software in tracking particles in a crawling cell.** (A) Examples of kymograph simulating a cell crawling and experiencing spreading generated with different SNR. (B) Average distance between found trajectories and input (simulated) trajectories as function of SNR; bars represent the standard deviation. (C) Average relative error in measured versus input particle velocity as a function of SNR; bars represent the standard deviation. (D) Line coverage: ratio between the total length of found trajectories and the total length of the input (simulated) trajectories as a function of SNR.
Data that might be obtained from a moving cell observed with DIC microscopy was also simulated and analyzed. In DIC images, the edges of objects are enhanced, improving contrast (12, 13). The algorithm detects local peak intensities, which also allows it to track edges in kymographs generated from DIC image sequences. Because of the interference nature of DIC microscopy, the local intensity maximum of an imaged object may not be positioned at exactly the physical position of its edge. Therefore, the algorithm only finds the relative edge position, which is less accurate than edges obtained with fluorescence microscopy. In the simulated DIC kymograph presented in Figure S12, the average error in edge localization is $0.83 \pm 0.83$ pixels (average ± standard deviation) and the relative error in velocity determination is $1.23 \pm 2.09\%$.

**Figure S12:** Example of a kymograph simulating a crawling cell observed with DIC microscopy. Blue lines are the trajectories found by the algorithm.
3.2.15 Simulation of bidirectional motion – performance with crossing particles

Bidirectional motion is often observed in intracellular transport (2, 14, 15, 16). Kymographs generated from microscopy image sequences obtained from these processes are difficult to analyze, because particles trajectories can cross and thus overlap. Furthermore, particles can be very numerous and their motion can vary stochastically from particle to particle and from time to time. The automated kymograph analysis algorithms described in this chapter have been designed to cope with these difficulties to provide straightforward access to the data richness hidden in the kymographs obtained from such experiments.

To test the applicability and accuracy of the program on this kind of data, two data sets were simulated: one with all particles moving only in one direction (Fig. S13A), and one with particles moving either in the forward or backward direction, creating about 100 crossing points in the kymographs (Fig. S13B). Stepping of the particles is in both cases stochastic: at each time point, a forward-moving particle steps randomly in the range of 0 to 5 pixels while a backward-moving particle makes a step in the same size range in the opposite direction. The particle and background intensities are Poisson distributed and varied in order to test a wide range of SNRs. The output of the algorithm is only slightly affected by particle crossings: even in kymographs displaying many crossing lines, the algorithm successfully and accurately detects individual trajectories running in both directions, with sub-pixel accuracy, even at very low SNR (Fig. S13C).

Compared to similar data displaying no crossing points, the loss of accuracy in detecting individual trajectories is limited to about a tenth of a pixel over the range of SNR tested (Fig. S13C). The ability of the algorithm to detect the presence of trajectories is not noticeably affected by the occurrence of crossing lines (Fig. S13D). These results are not surprising since Fourier filtering discriminates effectively between the different directions of motion in the kymograph: crossings are absent from the filtered kymograph used by the tracking algorithm and the performance of KymographDirect is therefore hardly affected if bidirectional motion occurs.
DISCUSSION

Kymographs are widely used by the cell biology and biophysics communities to visualize motion. Automated analysis tools for kymographs have been lacking, restricting researchers to manually extract motility parameters. In this chapter, a novel toolset for the generation and automated analysis of kymographs has been described, reliable even under severely crowded and low-SNR conditions. This toolset will enable microscopists to analyze their images with high accuracy, reliability and throughput, accelerating discoveries in cellular and intracellular dynamics.

Figure S13: Validation of the kymograph software using simulated data representing particles moving in two directions. (A) Example of a simulated kymograph containing crossing points due to particles moving in opposite directions. (B) Example of a kymograph simulating stepping particles moving only in the forward direction. (C) Average distance between found trajectories and input (simulated) trajectories as a function of SNR for bidirectional motion (black) and unidirectional motion (red); bars represent the standard deviation and the plain lines are exponential fits of the data (black, bidirectional motion, red, unidirectional motion). (D) Line coverage: ratio between the total length of a found trajectory and the total length of the input (simulated) trajectories as a function of SNR bidirectional motion (black) and unidirectional motion (red). The grey dashed line is a guide for the eye.
3.2.16 Acknowledgements

I would like to thank Jonathan M. Scholey (University of California, Davis) for the gift of the *C. elegans* strain expressing the kinesin motor OSM-3 tagged with GFP, as well as Jona Mijalkovic and Nuria Taberner for comments on the manuscript and software. This work was supported by a Vici fellowship from the Netherlands Organization for Scientific Research, Physics (NWO-N), by an ALW Open Programme grant from NWO, Earth and Life Sciences (NWO-ALW), an NWO Groot investment grant, and a grant from NanoNextNL.
REFERENCES


