



# Particle finder: a simple particle detection tool for continuous-flow systems

Raúl Fernández-Mateo<sup>1</sup> · Víctor Calero<sup>2</sup> · Pablo García-Sánchez<sup>3</sup> · Antonio Ramos<sup>3</sup> · Hywel Morgan<sup>1</sup>

Received: 6 October 2022 / Accepted: 11 January 2023 / Published online: 3 February 2023  
© The Author(s) 2023

## Abstract

We describe a user-friendly, open source software for single-particle detection/counting in a continuous-flow. The tool automatically processes video images of particles, including pre-conditioning, followed by size-based discrimination for independent detection of fluorescent and non-fluorescent particles of different sizes. This is done by interactive tuning of a reduced set of parameters that can be checked with a robust, real-time quality control of the original video files. The software provides a concentration distribution of the particles in the transverse direction of the fluid flow. The software is a versatile tool for many microfluidic applications and does not require expertise in image analysis.

**Keywords** Continuous-flow microfluidics · Particle detection · GUI · Particle counting software

## 1 Introduction

In continuous-flow microfluidics there is a common need for simple, rapid estimation of particle positions in the transverse direction of the fluid flow. The increasing amount of processed sample and volumetric throughput achieved in microfluidics demands for tools of particle detection which can process large amounts of information systematically.

This is specially critical in processes intended for Point-of-Care applications (Lakhera et al. 2022).

Areas of interest include particle focusing (Motosuke et al. 2013; Jia et al. 2015; Fernandez-Mateo et al. 2022) and particle fractionation caused by phenomena such as Deterministic Lateral Displacement (DLD) (Calero et al. 2019, 2020), pinched-flow fractionation (PFF) (Yamada et al. 2004; Pødenphant et al. 2015), viscoelastic flow (Liu et al. 2017; Tian et al. 2017) or electrokinetics (Tayebi et al. 2021; Emmerich et al. 2022), among others. For different populations of particles flowing simultaneously, differentiation is critical in order to estimate parameters such as fractionation efficiency or relative concentration (Thomas et al. 2017).

Often the methodology for performing this kind of analysis (summarised in Fig. 1) is not reported (Liang et al. 2010; Liu et al. 2017, 2018) or standardised, and in many occasions relies on manual counting of particles (Ho et al. 2021) or indirect measurements such as fluorescence intensity (Ho et al. 2020) and/or image stacking (Liang et al. 2010). Apart from a time-consuming task, these methods do not provide an estimate of the accuracy of the resulting measurements.

We describe a simple, open-source software tool based on MATLAB programming language (MathWorks, Natick, Massachusetts) for particle detection and classification. It estimates particles centre positions with sub-pixel precision and allows for real-time visualization of the results. It stores particle positions data from all analyzed frames and plots a histogram of particle counts in the transverse direction to

---

✉ Raúl Fernández-Mateo  
r.fernandez-mateo@soton.ac.uk

Víctor Calero  
victor.calero@inl.int

Pablo García-Sánchez  
pablogarcia@us.es

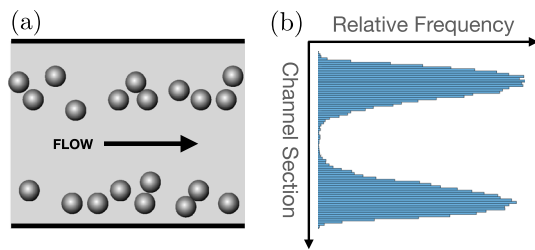
Antonio Ramos  
ramos@us.es

Hywel Morgan  
hm@ecs.soton.ac.uk

<sup>1</sup> School of Electronics and Computer Science, University of Southampton, University Street, Southampton SO17 1BJ, UK

<sup>2</sup> International Iberian Nanotechnology Laboratory (INL), Avenida Mestre José Veiga s/n, Braga 4710, Portugal

<sup>3</sup> Depto. Electrónica y Electromagnetismo. Facultad de Física, Universidad de Sevilla, Avda. Reina Mercedes s/n, Seville 41012, Spain

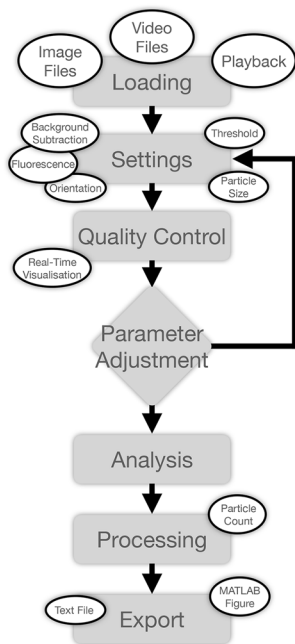


**Fig. 1** Example of **a** the initial images of particles flowing in a microfluidic channel and **b** the final histogram that bins the positions of the particles in the direction transverse to the flow

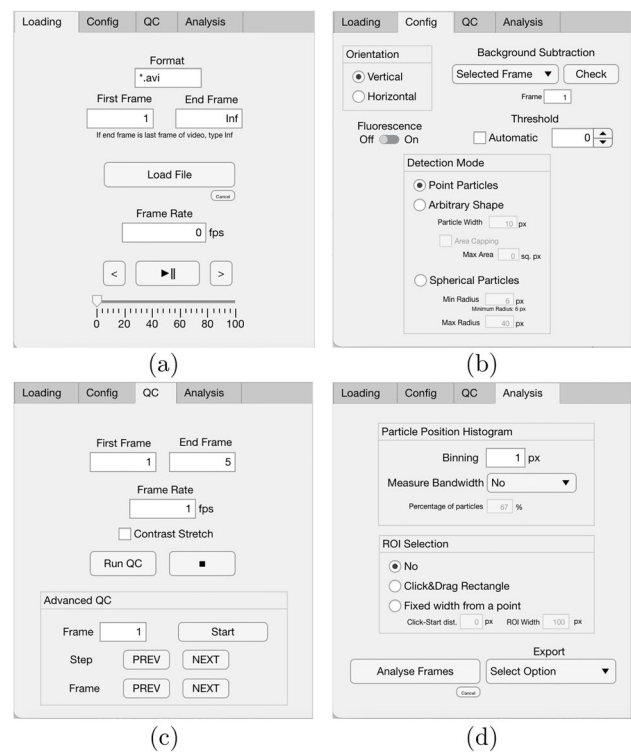
the fluid flow. The width of the resulting histograms can be determined in the software, and exporting options allow further processing of the data.

## 2 Running the GUI

The software general workflow is summarised in Fig. 2 and it is openly available at Fernandez-Mateo (2022). The Graphic User Interface (GUI) contains four tabs (shown in Fig. 3) which are designed to be simple and straightforward to use. Navigating through them in order will result in the



**Fig. 2** Workflow of the program while navigating through the tabs. After loading the Video or Image files selecting the first set of parameters, the results are displayed in real-time to allow the user to fine-tune the selection to optimise the image processing. The image or video file is then analysed without visualization to maximise the processing speed. The results can be exported to a MATLAB figure or a text file



**Fig. 3** The four tabs of the *Particle Finder* App, with the functions described in Sects. 2.1–2.4

detection, visualization and whole file analysis. Finally, the results may be exported to either a MATLAB figure or a text file with the raw data of transverse positions found for further post-processing of the results. Preliminary versions of the software have been tested to work flawlessly on different MATLAB versions (2019b to 2021b) under the main operating systems macOS X, Windows 10 and Ubuntu/LINUX. The final application has been packaged in the 2021b Runtime for Windows 10 operating system, and 2020b for macOS X. These are automatically added during the installation of the software and can be run without the necessity of having MATLAB installed.

### 2.1 Loading and playback

The software interactively imports data from local folders. This may be video or a sequence of image files from the *Load* (see Fig. 3a). Supported common image file formats are *.bmp*, *.jpeg*, *.png*, and *.tiff*. Supported video formats are *.avi*, *.mp4*, *.m4v* and *.mj2*, but these may depend on the operating system and MATLAB Runtime version installed with the software. When loading video files, the user can select the initial and final frames to be imported.

Once loaded, the frames appear in the graphical screen located at the right-hand side of the GUI window. The software provides play/pause of frames at original or selected

rates, displays the images frame by frame, or move to a selected frame, as shown in Fig. 3a.

## 2.2 Configuration of parameters

The specification of the general properties of the frames, as well as the particles to be detected is done in the *Config* tab of the GUI (Fig. 3b). The user selects the direction of flow on the frame sequence, and also whether the particles are fluorescent or non-fluorescent (i.e. shades in bright-field images).

The next step is background correction. Five different background subtraction options are available depending on the relationship between the target particles and the environment.

**Selected frame**—A frame without target particles is used as a reference to identify unimportant and static features in the images. The frame number is input below the drop-down menu.

**Last frame**—If the concentration of particles is low enough so that frames without particles are found, these can be used to set a dynamic background subtraction. This regenerates each time a frame with no particles is detected. This feature is convenient if images have slight variations in background illumination during an experiment. In order to start this process, a seed frame is selected to generate an initial background. The user inputs the seed frame number in the same location as the “selected frame” option below the drop down menu.

**AVG**—The software averages the intensity of each pixel position over all frames. This is the most versatile type of subtraction since it may be used for fluorescent or non-fluorescent particles.

**MIN and MAX**—When MIN is selected, the software returns the minimum intensity found in each pixel position for all the evaluated frames. This is normally the most robust background when evaluating fluorescent particles. In the same way, the MAX option selects the maximum intensity of all frames for each pixel position, which is most convenient for non-fluorescent objects.

For a simple robust platform, no correlation of particles is performed between frames. The optimal procedure for identifying objects is binarization of the images to completely isolate the objects and an important parameter in this process is the threshold. The threshold is the intensity value above which the software allocates an intensity value of 1 to a certain pixel position. Else, the value 0 is assigned.

The Particle Finder automatically designates a threshold estimation for the images based on the Otsu method (Otsu 1979). To improve the estimation, the fluorescent nature of the particles and the background subtraction method should be previously selected. The automatically chosen value may be further optimised by the user. To tune the value of the

threshold, it is important to consider that the threshold is the intensity value which marks the difference between the targeted particles and the background noise, once the selected background is subtracted from the original frame. The magnitude of this intensity value is measured in the scale of the bit depth of the frames (for example, from 0 to 255 in 8-bit depth images).

The last stage is the detection mode, which is the criterion the software uses to discern a target particle between all pixels that have passed the threshold and noise reduction cuts. Particle Finder provides three different detection modes depending on the particle sizes and shape.

First the *Point Particles* mode detects particles that occupy single pixels in the frames. To achieve this the software makes use of the MATLAB custom function *find*, which locates the non-zero elements of a given matrix (here images are represented as matrices). This function is outstandingly fast in detecting particles when compared to the other two detection modes. Also, when this detection mode encounters particles that extend more than one pixel, it automatically removes them from the count. This feature may also be used for a size-based discrimination.

The second detection mode, *Arbitrary Shape* isolates particles of arbitrary size whose characteristic length is given by input parameter *Particle Width*, expressed in pixels. The function is *regionprops*, in the image processing toolbox of MATLAB. This function is flexible in terms of the size of the detected particles and may select sizes much larger than specified. Therefore an option to select a maximum detected area of the objects is included (in square pixels). The versatility of this detection mode lies in its ability to detect arbitrary shapes of any size. The particle position is finally given by the centroid of the detected distribution.

The last available detection mode, *Spherical Particles*, detects circles with radii ranging between two given input values; with a minimum value of 6 pixels in order to consistently discern circular patterns. If the apparent size of the spherical particles is below 6 pixels, the second detection option may be used. The software uses the function *imfindcircles*. The power of this algorithm lies in the ability to discern different particles even when they are aggregated due to the condition of spherical shape, while ignoring non-spherical shapes.

Finally, we note that the apparent size in pixels may vary depending on threshold conditions—higher thresholds could cause a reduction of the apparent size of the objects, and this should be considered when selecting the size parameters. However, as the histograms are built on the particle number count and not their intensity integration, the final histogram count is independent of the value set for the threshold as long as it allows the identification of the targeted particles. That is, if two similar but not equal thresholds that properly identify the targeted particles are found (which is often

possible when particles are clearly distinguishable from the background), then the resulting histograms are identical.

### 2.3 Quality control and visualization

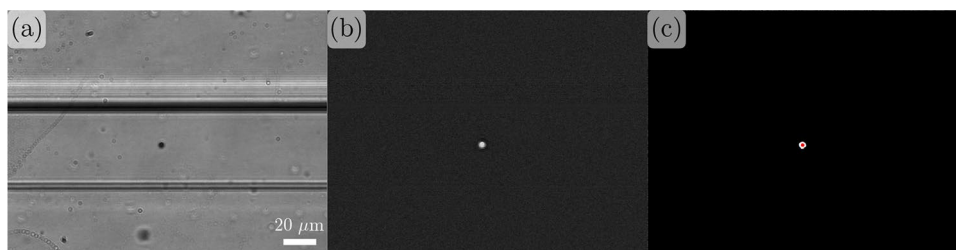
Once all parameters have been set, the next tab *QC* shown in Fig. 3c previews in real time the accuracy of the particle detection for the chosen parameters. This provides a means of fine-tuning the parameters without processing all the frames.

For this purpose, the user inputs a range of frames to preview and the frame rate. Pressing *Run QC* button, the software displays the original frames with the centres of the detected objects marked with red dots. The visualization can be stopped at anytime by pressing the stop “■” button. For the case where the background is similar in intensity to the detected objects making them difficult to distinguish, the *Contrast Stretch* option labels the maximum and minimum value of intensity for each frame.

Finally, for the case where the detection parameters are particularly complex to optimise, the tool *Advanced QC* provides a feature for the user to explore each step in the particle detection for the selected settings and detection mode. Pressing *Start* takes the user to the first unmodified frame selected. From this point, it is possible to navigate through the different steps of the detection process i.e. background subtraction, binarization, noise reduction and particle identification. At any stage it is possible to move to the adjacent frames at the same detection step.

### 2.4 Complete analysis, data exploration and exporting

The detection process is summarised as follows: First, the background is subtracted from each frame; next, the resulting image is binarized where the selected threshold and noise reduction filters are automatically applied. Finally the selected detection mode is applied. This process is visually described in Fig. 4.



**Fig. 4** The three primary steps taken by Particle Finder when detecting particles. **a** Original unmodified frame, where a non-fluorescent particle is seen in the center of the channel together with a background with uneven illumination. **b** Frame inverted and background

Particle Finder is designed to analyse a large number of frames in a single session. The tool was tested with videos up to 3000 frames. To reduce the time taken to analyse a large set of images, pressing the *Analyse Video* button in the *Analysis* tab (Fig. 3d), hides the displayed images.

To target the required objects and optimise the computational effort, this software can also select a Region of Interest (ROI). There are two different ways of selecting the ROI: The first is to click-and-drag with the cursor a rectangle of arbitrary shape. An alternative option allows the user to make a more accurate selection of the ROI by adjusting the width of the ROI rectangle in the direction of the fluid flow.

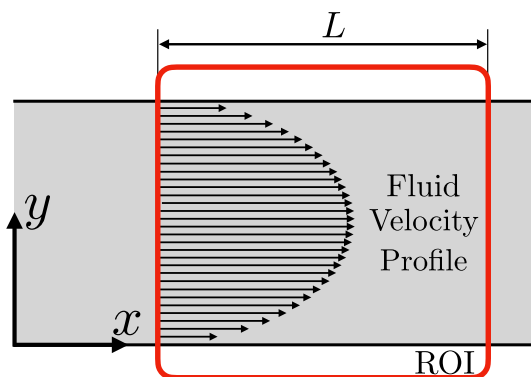
An important step is to configure how the data is processed. Particle Finder stores the transverse positions of the objects independent of their longitudinal position within the ROI. This is the most direct way for particle counting. Although this counting method does not reflect the number of particles detected, since one particle may appear in two or more consecutive frames as they flow, for a large enough number of frames the relative frequency of appearance across the transverse direction to the fluid motion represents the relative concentration of particles in this direction. Therefore, the information that can be extracted from the software meets the requirements for the type of analysis described in the literature.

Following the notation described in Fig. 5, in which the transverse direction to the fluid flow is denoted as  $\hat{y}$  and the longitudinal direction as  $\hat{x}$ , we see that for an initial random concentration of particles in  $\hat{x}$ , and on average a spacing with distance  $\bar{x}$ , the average time  $\bar{t}$  it takes for the next particle to cross the same  $y$  position is  $\bar{t} = \bar{x}/u(y)$ . Here,  $u(y)$  is the fluid velocity profile across the section of the channel. Therefore, the number of particles crossing into the ROI on average per unit time  $N_t$  at position  $y$  is

$$N_t(y) = \frac{u(y)}{\bar{x}}. \quad (1)$$

This means that the number of particles crossing into the ROI does not properly represent the relative concentration of

subtracted. **c** Binarisation and noise reduction filters applied. The red dot marks the position of the particle in the frame assigned by the software. The example is for a 3  $\mu\text{m}$  bead in a 50  $\mu\text{m}$  wide channel



**Fig. 5** Illustration of the parabolic profile for a straight channel with a region of interest (ROI) selected. If the concentration of particles is homogeneous along the x direction, for a fixed frame rate the number of counts is proportional to the particle concentration within the channel cross section

particles in the flow, but is biased by the particular velocity profile of the fluid. However, the software counts the number of times each particle appears in the ROI.

If the ROI has a length  $L$  in the  $\hat{x}$  direction and frames are taken at a certain rate  $f_r$ , the number of times a particle appears in the ROI on average  $N_f$  is

$$N_f(y) = \frac{f_r L}{u(y)}, \tag{2}$$

so for image sets recorded for long enough time, the total number of counts  $C$  as a function of the  $\hat{y}$  direction is the number of particles flowing through the ROI in that period, times the number of frames the particle spends in the ROI,

$$C(y) = N_r N_f \Delta t = f_r L \frac{\Delta t}{\bar{x}}. \tag{3}$$

This demonstrates that the number of counts is not influenced by the velocity profile of the fluid flow but responds to the real relative concentration of particles in the transverse direction, as none of the parameters in (3) depends on  $y$ . The reason behind this result is that in the  $y$  positions where the fluid flow is faster, the particles cross the ROI more frequently, but as they go faster they appear in fewer frames. These two facts compensate each other to obtain counts proportional to the actual particle concentration in the perpendicular direction to the fluid flow  $\hat{y}$ .

Finally, after analysing all frames, the software displays a histogram binning particle positions in the  $\hat{y}$  direction. The user can select the width of the bins in pixels. The software exports the histogram into a MATLAB figure for further processing or as raw data in a text file.

Particle Finder also offers one post-processing option: measurement of the width of particle bands, i.e. the total width of the beam of particles detected or a selected

percentage of the concentration. This measurement can be done in the *Analysis* tab of the software by choosing in the *Measure Bandwidth* selector either *From Median* if we want to count the percentage of particles from the median of the distribution, or *From Ends* if the count is to be made from the ends of the distribution.

### 3 Example

This section illustrates the operation of Particle Finder with an example video where two distinct particle populations flow in a straight channel. This video is openly available in the Supplementary Information.

In this case, particles flow in a  $50 \times 50 \mu\text{m}$  square cross-section straight microfluidic channel. The particle sample contains two distinct populations: the small particles are 500 nm diameter fluorescent beads, while the larger particles are  $2 \mu\text{m}$  in diameter and non-fluorescent. They are suspended in an aqueous solution flowing at  $7.5 \mu\text{l/h}$ .

After loading all the available frames of the .avi file, we start the independent detection of both populations. For the case of the 500 nm beads, we select the “Horizontal” direction of the flow, as in the video the particles are flowing from left to right. This population is fluorescent, so the particles are brighter than the background. This is indicated by selecting the “On” position in the fluorescence switch. Finally we choose the minimum projection of the stack of frames as the background, done by selecting the “MIN” option in the drop-down menu.

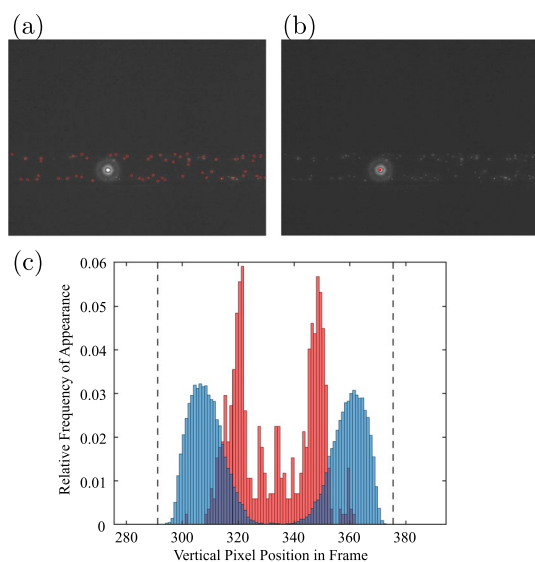
All these preconditioning options allow detection of the smaller particles when selecting a binarization threshold of 8 with the second detection mode “Arbitrary Shape”. As seen in the example video, these particles have a typical length of 8 pixels. To avoid counting the  $2 \mu\text{m}$  particles, we select the “Area Capping” checkbox to avoid particles above a certain area (200 square pixels) being counted.

Moving to the next tab into the QC section, we observe detection of the 500 nm beads, while the bigger particles are not counted. An example frame showing the two populations but with only smaller particles counted is presented in Fig. 6a. Finally, analysing the complete set of frames results in the blue bars of the histogram in Fig. 6c.

Detection of the larger  $2 \mu\text{m}$  particles requires the third detection mode. Choosing 10 pixels as the minimum particle radius and 40 pixels as the maximum, the smaller beads will be ignored while the targeted particles are correctly identified. To compensate for the brighter objects, the binarization threshold is set as 35. This is seen in the example frame shown in Fig. 6b.

The final relative frequency for the larger particles is shown by the red bars in the histogram of Fig. 6c. For creating this figure, the results from both analysis have





**Fig. 6** Example of different populations of particles flowing simultaneously in a 50  $\mu\text{m}$ -wide channel. **a** 500 nm particles detect by the QC in the software. **b** Larger 2  $\mu\text{m}$  particles detected whilst ignoring the smaller particles. **c** Superimposed histogram of particle distribution in the ROI for 500 nm particles (blue bars) and 2  $\mu\text{m}$  particles (red bars). The position of the channel walls are marked by the dashed vertical lines

been separately exported and combined in a single plot using MATLAB.

## 4 Conclusions

We have described a computational tool that provides users with no experience in image analysis with a user-friendly software for detection and analysis of particles in a flow. This is particularly useful in the case of microfluidic applications.

Preliminary versions of this software have already been used for previous publications (Fernandez-Mateo et al. 2022; Gillams et al. 2022) with different flow profiles, microchannel geometries and particle fluorescent intensities and size. The code is robust and versatile, and together with a simple GUI, it is very practical and easy to use for applications in particle analysis for microfluidics.

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1007/s10404-023-02626-7>.

**Data availability statement** The data supporting this publication is openly available from the University of Southampton repository at <https://doi.org/10.5258/SOTON/D2503>.

## Declarations

**Conflict of interest** The authors declare no conflict of interest

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Calero V, Garcia-Sanchez P, Honrado C, Ramos A, Morgan H (2019) Ac electrokinetic biased deterministic lateral displacement for tunable particle separation. *Lab Chip* 19(8):1386–1396. <https://doi.org/10.1039/C8LC01416G>
- Calero V, Garcia-Sanchez P, Ramos A, Morgan H (2020) Electrokinetic biased deterministic lateral displacement: scaling analysis and simulations. *J Chromatogr A* 461151. <https://doi.org/10.1016/j.chroma.2020.461151>
- Emmerich ME, Sinnigen A-S, Neubauer P, Birkholz M (2022) Dielectrophoretic separation of blood cells. *Biomed Microdevices* 24(3):1–19. <https://doi.org/10.1007/s10544-022-00623-1>
- Fernandez-Mateo R (2022) Particle finder: detection tool for continuous-flow systems. MATLAB Central File Exchange. <https://uk.mathworks.com/matlabcentral/fileexchange/117865-particle-finder-detection-tool-for-continuous-flow-systems> (Matlab Central File Exchange). Accessed 22 Sept 2022
- Fernandez-Mateo R, Calero V, Morgan H, Garcia-Sanchez P, Ramos A (2022) Wall repulsion of charged colloidal particles during electrophoresis in microfluidic channels. *Phys Rev Lett* 128:074501. <https://doi.org/10.1103/PhysRevLett.128.074501>
- Gillams RJ, Calero V, Fernandez-Mateo R, Morgan H (2022) Electrokinetic deterministic lateral displacement for fractionation of vesicles and nano-particles. *Lab Chip*. <https://doi.org/10.1039/D2LC00583B>
- Ho BD, Beech JP, Tegenfeldt JO (2020) Charge-based separation of micro- and nanoparticles. *Micromachines* 11(11):1014. <https://doi.org/10.3390/mi11111014>
- Ho BD, Beech JP, Tegenfeldt JO (2021) Cell sorting using electrokinetic deterministic lateral displacement. *Micromachines* 12(30). <https://doi.org/10.3390/mi12010030>
- Jia Y, Ren Y, Jiang H (2015) Continuous-flow focusing of micro-particles using induced-charge electroosmosis in a microfluidic device with 3d apdms electrodes. *RSC Adv* 5(82):66602–66610. <https://doi.org/10.1039/C5RA14854E>
- Lakhera P, Chaudhary V, Bhardwaj B, Kumar P, Kumar S (2022) Development and recent advancement in microfluidics for point of care biosensor applications: a review. *Biosens Bioelectron X* 11:100218. <https://doi.org/10.1016/j.biosx.2022.100218>
- Liang L, Ai Y, Zhu J, Qian S, Xuan X (2010) Wall-induced lateral migration in particle electrophoresis through a rectangular microchannel. *J Colloid Interface Sci* 347(1):142–146. <https://doi.org/10.1016/j.jcis.2010.03.039>
- Liang L, Qian S, Xuan X (2010) Three-dimensional electrokinetic particle focusing in a rectangular microchannel. *J Colloid Interface Sci* 350(1):377–379. <https://doi.org/10.1016/j.jcis.2010.06.067>
- Liu C, Guo J, Tian F, Yang N, Yan F, Ding Y, Sun J (2017) Field-free isolation of exosomes from extracellular vesicles by

- microfluidic viscoelastic flows. *ACS Nano* 11(7):6968–6976. <https://doi.org/10.1021/acsnano.7b02277>
- Liu Z, Li D, Saffarian M, Tzeng T-R, Song Y, Pan X, Xuan X (2018) Revisit of wall-induced lateral migration in particle electrophoresis through a straight rectangular microchannel: Effects of particle zeta potential. *Electrophoresis* 1–6. <https://doi.org/10.1002/elps.201800198>
- Liu Z, Li D, Song Y, Pan X, Li D, Xuan X (2017) Surface-conduction enhanced dielectrophoretic-like particle migration in electric-field driven fluid flow through a straight rectangular microchannel. *Phys Fluids* 29(10):102001. <https://doi.org/10.1063/1.4996191>
- Motosuke M, Yamasaki K, Ishida A, Toki H, Honami S (2013) Improved particle concentration by cascade ac electroosmotic flow. *Microfluidics Nanofluidics* 14(6):1021–1030. <https://doi.org/10.1007/s10404-012-1109-1>
- Otsu N (1979) A threshold selection method from gray-level histograms. *IEEE Trans Syst Man Cybern* 9(1):62–66. <https://doi.org/10.1109/TSMC.1979.4310076>
- Pødenphant M, Ashley N, Koprowska K, Mir KU, Zalkovskij M, Bilenberg B, Marie R (2015) Separation of cancer cells from white blood cells by pinched flow fractionation. *Lab Chip* 15(24):4598–4606. <https://doi.org/10.1039/C5LC01014D>
- Tayebi M, Yang D, Collins DJ, Ai Y (2021) Deterministic sorting of submicrometer particles and extracellular vesicles using a combined electric and acoustic field. *Nano Lett* 21(16):6835–6842. <https://doi.org/10.1021/acs.nanolett.1c01827>
- Thomas C, Lu X, Todd A, Raval Y, Tzeng T-R, Song Y, Xuan X (2017) Charge-based separation of particles and cells with similar sizes via the wall-induced electrical lift. *Electrophoresis* 18:320–326. <https://doi.org/10.1002/elps.201600284>
- Tian F, Zhang W, Cai L, Li S, Hu G, Cong Y, Sun J (2017) Microfluidic co-flow of Newtonian and viscoelastic fluids for high-resolution separation of microparticles. *Lab Chip* 17(18):3078–3085. <https://doi.org/10.1039/C7LC00671C>
- Yamada M, Nakashima M, Seki M (2004) Pinched flow fractionation: Continuous size separation of particles utilizing a laminar flow profile in a pinched microchannel. *Anal Chem* 76:5465–5471. <https://doi.org/10.1021/ac049863r>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.