Title

Advanced Optofluidic Devices and Signal Analysis for Point-of-Care Applications

Author

Ganjalizadeh, Vahid

Publication Date

2023

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

ADVANCED OPTOFLUIDIC DEVICES AND SIGNAL ANALYSIS FOR POINT-OF-CARE APPLICATIONS

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

ELECTRICAL AND COMPUTER ENGINEERING

by

Vahid Ganjalizadeh

June 2023

The Dissertation of Vahid Ganjalizadeh
is approved:

Professor Holger Schmidt, Chair

Professor Hamid Sadjadpour

Professor Marco Rolandi

Peter F. Biehl
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Advanced optofluidic devices and signal analysis for point-of-care applications

by

Vahid Ganjalizadeh

Optofluidics as a relatively new field of biomedical detections has become an attractive technology that combines integrated photonics and microfluidics. This brings together the best of optical sensing and microfluidic media for biological fluidic compatibility to attain high sensitivity. Integrated designs with compact and portable footprints are a unique feature of optofluidics which makes it a great candidate for point-of-care (POC) applications. Even though there have been outstanding results in terms of sensitivity and multiplexing capability reported in the past couple of decades, there is still significant room for innovation to boost performance. The dynamic environment of a microfluidic channel causes ultrasensitive photonic sensing challenges because of many variables changing from chip to chip, sample to sample, and even in the course of a measurement. First, we explore the potential of incorporating microfluidic techniques, specifically, 3D hydrodynamic focusing (3DHDF) to minimize these variations within the experiment. With the help of 3DHDF, we are able to focus the biological sample fluid into a narrow stream where all of the flowing target particles experience similar light-matter interaction. This, of course, requires a change in the chip design and involves a more complicated fabrication process. Therefore, other alternative approaches were considered. A robust powerful event detection framework is introduced. It suits the multiplexed real-time pathogen detection in the presence of possible background and noise signals as well as signal variations. The use of continuous wavelet transform (CWT) in this algorithm alongside the custom wavelet design takes advantage of multi-scale analysis to detect events across scales. Next, in a different approach, a machine learning framework with a convolutional neural network (CNN) is used for classifying PCWA-detected events very efficiently and fast. The framework is further pushed in terms of compatibility with current trends toward edge devices by a successful implementation of the full real time event detection and classification on an edge device is demonstrated. This proves the capacity of edge computing devices for a broad range of applications such as portable ultrasensitive biomedical diagnosis instruments. Furthermore, the current multiplexing concept is expanded towards the high concentration range of operation. In this regime (also called analog regime), individual events are not detectable anymore and the measurement relies on the analog signals recorded from highly concentrated solutions. Wavelength division multiplexing (WDM) was previously used successfully for multiplexed detection of individual events by encoding spectral information into the time domain. Here, a hybrid adaptive scheme adds time division multiplexing into this recipe to increase the dynamic range of the optofluidic biosensor. The results show significant (four orders of magnitude) expansion of the multiplexed detection at higher concentrations while preserving the already established performance at lower concentrations.

Acknowledgments

I thank my caring and encouraging family, especially my dear parents who with their endless support, I had the chance to live, learn, and pursue my dreams. My elder siblings were my first and best mentors I started playing and learning from childhood and they never stopped helping me whenever I needed them.

My second family, here in Santa Cruz, never let me miss thousands of miles of distance from my home and always encouraged and cherished me throughout my Ph.D. life. I'm especially thankful to my advisor Prof. Holger Schmidt for his professional mentorship and patience with my progress. His support from the very beginning made it possible for me to join the Applied Optics lab, and learn and work with so many amazing people. In moments of trouble, he was the first person to help and solve the problems. Gratefully I acknowledge my committee members, Prof. Marco Rolandi and Prof. Hamid Sadjadpour, for their teaching, advice and suggestions which were very important in finishing my Ph.D.

Last but not least, I'm grateful to have amazing friends at UCSC to explore the amazing places, build memories and share jokes and meals with.

Chapter 1

Introduction

Disease detection is an essential process in the medical field that allows for the early identification and treatment of various illnesses. Point-of-care (POC) testing has emerged as an effective means of disease detection, particularly in resource-limited settings where access to sophisticated laboratory equipment is limited. The development of portable, easy-to-use, and low-cost diagnostic tools has become increasingly important in recent years, especially after the global challenges as a result of SARS-CoV2 since late 2019.

Signal processing and data analysis techniques play a critical role in disease detection. In the case of POC diagnosis, these techniques help to ensure the accuracy and reliability of test results. As diagnostic tests become more complex, the need for advanced signal processing and data analysis methods becomes more pressing. Therefore, it is important to continue improving these techniques to ensure accurate and reliable diagnoses.

Optofluidics is a rapidly growing field that combines optics and microfluidics to develop novel biosensors. It provides an excellent platform for the development of point-of-care instruments for disease detection. One of the key advantages of optofluidics is its ability to integrate multiple components such as light sources, detectors, and microfluidic channels on a single chip, thereby enabling the development of miniaturized and portable biosensors. Recent advancements in chip design, particularly the use of 3D hydrodynamic focusing [1, 2, 3], have resulted in significant improvements in the performance of optofluidic biosensors. This approach has been shown to enhance the sensitivity and selectivity of biosensors while reducing the complexity of the device. The ability to improve the performance of biosensors is critical to their adoption in clinical settings, particularly for the detection of diseases with low biomarker concentrations.

Recent developments in machine learning, particularly the emergence of edge devices, have the potential to revolutionize optofluidic biosensors. These devices have the ability to perform complex data analysis tasks on the device itself, thereby reducing the need for large and complex data analysis systems. Machine learning algorithms can be used to analyze data generated by biosensors and provide real-time feedback on the health status of patients. As the field continues to evolve, it is likely that we will see further developments in these areas, which will continue to improve the accuracy and reliability of diagnostic tests for public health benefits.

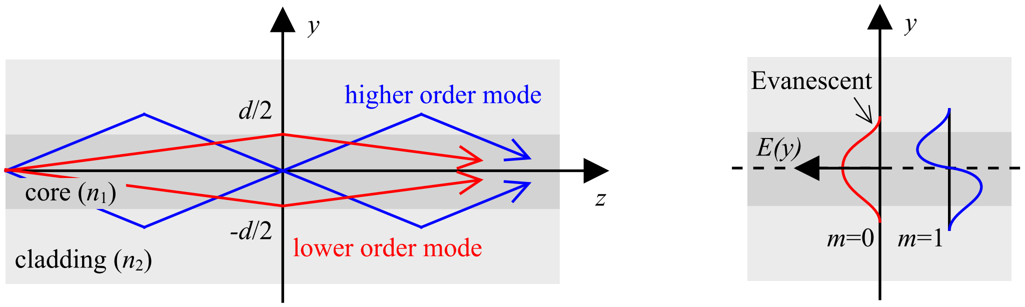


Figure 1.1: Slab waveguide with TIR guiding and mode field profile.

1.1 Optical waveguides

1.1.1 Slab waveguides

Slab waveguides as one of the simplest optical waveguides, guide light by total internal reflection (TIR). Fig. 1.1 shows a symmetric slab waveguide with lower and higher mode examples propagating using rays. Mode field profiles of the same rays are illustrated on the right indicating the oscillating behavior of the field (here electric field E_y) with increased evanescent wave penetrating the cladding layer (refractive index n_2) in the higher order mode.

Depending on the angle of incidence for the light, the constructive interference can be calculated based on the phase difference of incoming and reflected wavefronts from the boundary. Thus, for wave propagation in layer (n_1) along the guide we need

$$k_1 \left[2d \cdot cos\theta\right] - 2\phi = m(2\pi) \tag{1.1}$$

where $k_1 = kn_1 = 2\pi n_1/\lambda_1$, $d = 2a$ is slab thickness, k and λ are the free space

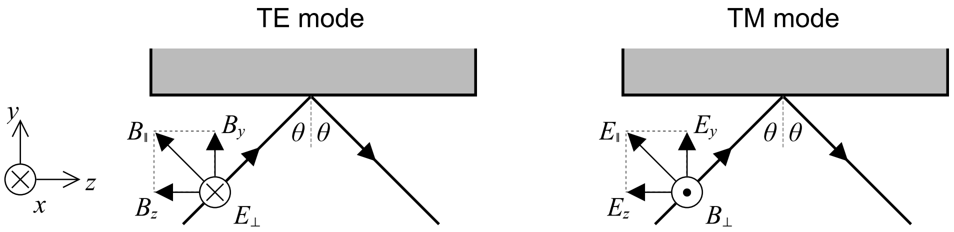


Figure 1.2: Transverse electric field (TE) and transverse magnetic field (TM) modes.

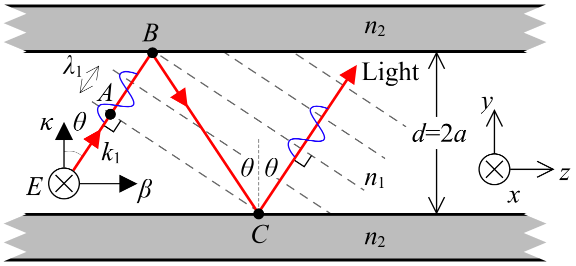


Figure 1.3: Light traveling with constructive interference.

wavevector and wavelength (see Fig. 1.3). ϕ is the phase change, thus is for certain θ and ϕ, the equation is valid for integer values of m. In order to find θ_m and ϕ_m values for a corresponding m we divide Eq. 1.1 by 2

$$\left[\frac{2\pi n_1(2a)}{\lambda}\right] cos\theta_m - \phi_m = m\pi \qquad (1.2)$$

Eq. 1.2 is called waveguide mode condition and indicates that ϕ_m is a function of incident angle θ_m. Fig. 1.3 shows possible modes in terms of electric or magnetic field direction with respect to the plane of incidence (here the plane of incidence is the paper).

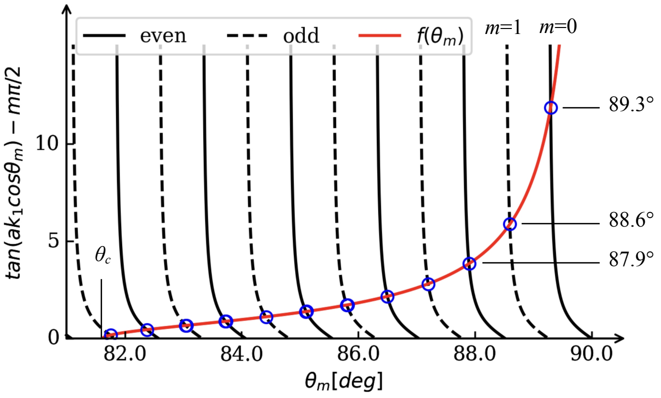


Figure 1.4: Modes in the planar dielectric waveguide.

1.1.2 Waveguide modes

Guiding modes in a slab waveguide can be determined by solving the equation where waveguide condition and Fresnel's equation $f(\theta_m)$ are equal

$$\begin{aligned} k_1\left[2d \cdot cos\theta\right] - 2\phi = f(\theta_m) \\ f(\theta_m) = \frac{\sqrt{\sin^2\theta_m - \left(\frac{n_2}{n_1}\right)^2}}{\cos\theta_m} \end{aligned} \tag{1.3}$$

1.1.3 Multimode interference (MMI) waveguides

A multimode interference (MMI) waveguide is an optical waveguide capable of guiding light via multiple modes, meaning that the physical dimensions should be big enough to contain several higher-order lateral modes. The main application of MMI waveguides is to split power and wavelength demultiplexing (WDM). For a step-index waveguide with width (w_m) and effective refractive indices of n_c and n_{cl} for the core and the cladding parts respectively, there are M supported lateral modes determined by the V number defined as

$$\begin{aligned} V &= \frac{\pi}{\lambda} w_m \sqrt{{n_c}^2 - {c_{cl}}^2} \\ M &= int\left(\frac{2V}{\pi}\right) + 1 \end{aligned} \tag{1.4}$$

An MMI pattern consists of multiple peaks and valleys in between representing the bright and dark regions of the interference pattern. This has been explained in-depth in [4]. Optofluidic biosensor chips are designed such that the MMI pattern formed at the excitation region inside the analyte channel shows different integer numbers of peaks for the different wavelengths of the input light. The following equation relates the mentioned variables in a multimode waveguide

$$N\lambda = \frac{n_c w_{eff}^2}{L} = cte \tag{1.5}$$

n_c, w_{eff}, and L are the core refractive index, effective width and length of the MMI waveguide respectively. If we neglect the dispersion of the waveguide, the right-hand side of the equation is a constant at any given wavelength, therefore the left-hand term should remain constant. This is the key to choosing the design parameters for our optofluidic sensors. Numerical simulations are used to predict interference patterns much more accurately based on the waveguide properties and dimensions. Examples of simulation results for two different wavelengths are shown in Fig. 1.4 (top) in which the waveguides are simulated based on the anti-resonant reflecting optical waveguides (ARROW) [5, 6, 7].

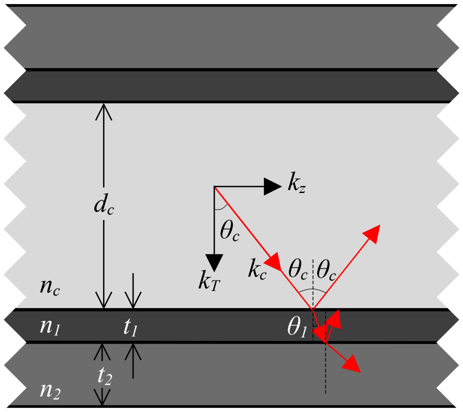


Figure 1.5: ARROW layers.

1.1.4 ARROW (anti-resonant reflecting optical waveguide)

Optofluidic biosensors rely on exciting fluorescently-tagged targets and collect the emission effectively. The fluidic medium for carrying these targets is usually water with a refractive index of $n_c = 1.33$. This low value of the refractive index in comparison to the high index values of SiO_2 ($n \approx 1.47$) makes it impossible to guide emission light inside the liquid-core channel via TIR.

ARROW waveguide solves this problem by forming an anti-resonant reflecting layer underneath the liquid channel (Fig. 1.5). A stack of alternating layers of low and high refractive indices (n_1, n_2), the thicknesses of each layer t_i can be calculated by

$$t_i = \frac{(2N-1)\lambda}{4n_i\sqrt{1 - \frac{n_c^2}{n_i^2} + \frac{\lambda^2}{4n_i^2 d_c^2}}}, \qquad N = 1, 2, 3, \ldots \tag{1.6}$$

This equation is valid if $n_1 > n_2, n_c$ so that the total phase shift from reflections becomes $\phi_1 + \phi_2 = 2\pi$. In the case of optofluidic devices, the thickness of the core layer

(d_c) is bigger than the ARROW layers, so $n_i^2 - n_c^2 \gg \frac{\lambda^2}{4d_c^2}$, then Eq. 1.6 can be simplified as

$$t_i = \frac{(2N-1)\lambda}{4\sqrt{n_i^2 - n_c^2}} \tag{1.7}$$

1.2 Optofluidic MMI devices

The principle of the optofluidic biosensors studied in this thesis is based on the MMI pattern to excite flowing particles inside an analyte channel with different excitation patterns. The patterns include spectral information of excitation wavelengths, thus encoding unique fingerprints for carefully tagged target molecules. Fig. 1.6 illustrates the concept of utilizing an MMI waveguide designed with carefully chosen dimensions to create an integer number of spots also known as multi-spot excitation (MSE) at selected wavelengths inside the analyte channel.

The integer number of peaks adds redundancy to the events to help with the confidence in the event detection (repeating pattern distinguishable from random background noise). Additionally, different number of peaks (from Eq. 1.5) in the collected signal (via a collection waveguide-CWG) helps us to distinguish which kind of target created the signals, hence providing multiplexing capability.

Characterizing the MSE is crucial to improve the signal quality of the biosensor. This can be done by taking a fluorescence microscopy image from the analyte channel when filled with a fluorescent dye and excited by a proper wavelength. Fig. 1.7 (left)

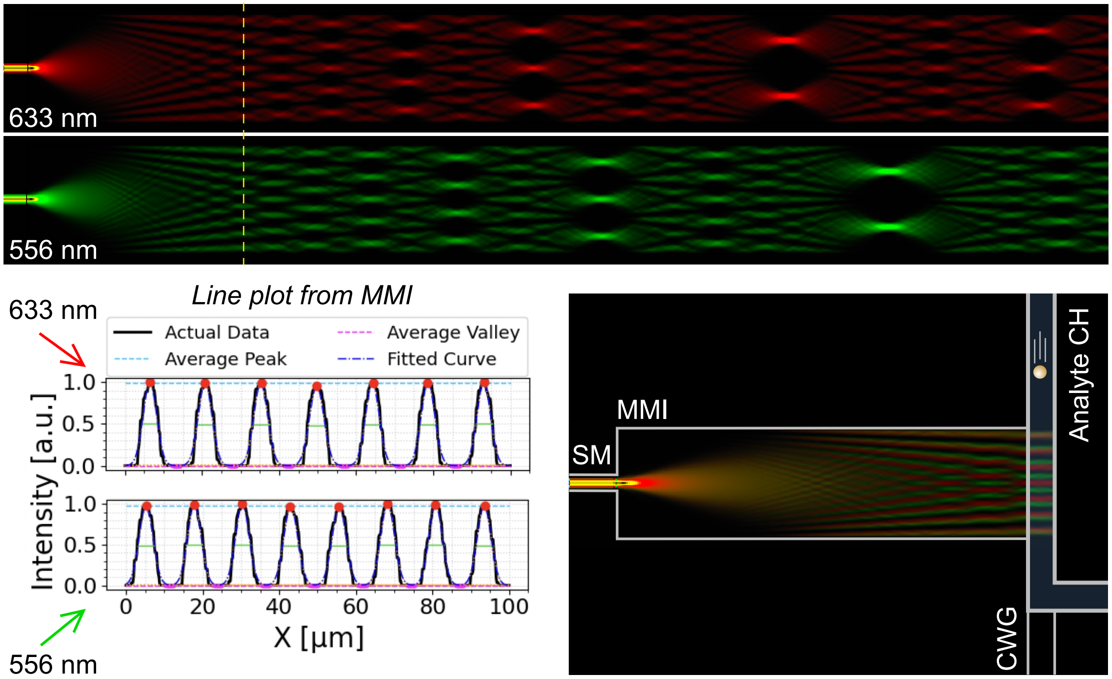


Figure 1.6: Multimode interference and principle of MMI biosensor.

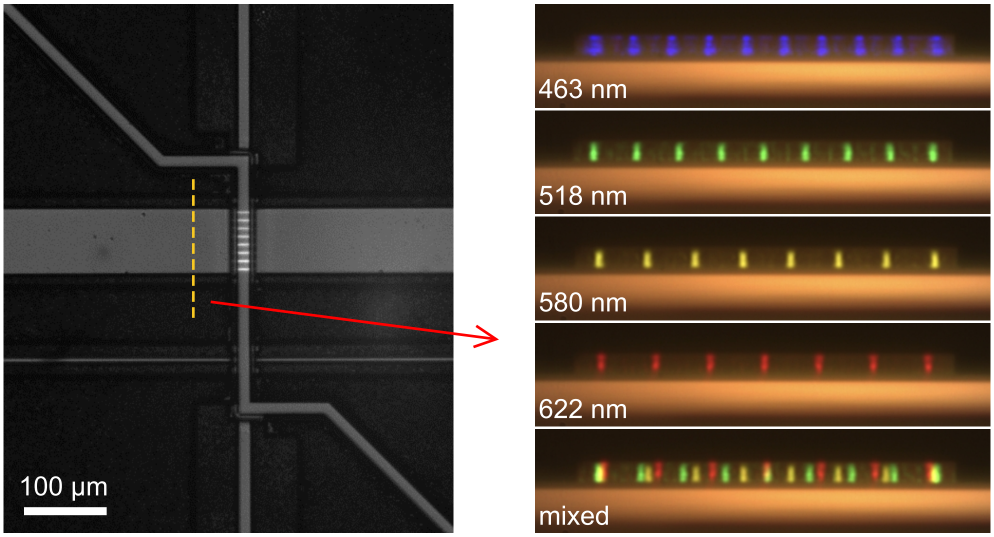


Figure 1.7: MSE patterns inside analyte channel (left) and at the facet (right).

is an example image taken from an ARROW device filled with a 25% quantum dot solution in deionized (DI) water. The clear bright and dark stripes are visible in the analyte channel with 8 distinct bright lines at $\lambda = 556$ nm.

When we cleave the chip somewhere before the channel, we can also look at the facet and analyze the MSE pattern. Fig. 1.7 shows multiple images taken from the facet of an ARROW chip at different wavelengths. Hunting the right wavelength for a distinct number of spots might become a challenge as the location of the cleaving can vary, therefore we developed a technique to ease this process. In Fig. 1.8 the steps of automated characterization of the MSE pattern are shown. A tunable laser is butt-coupled into the single-mode waveguide in an ARROW chip via a single-mode optical fiber and scans through a range of wavelengths. On the other end, a microscope objective lens is focused on the facet and a CMOS camera captures the image. The developed

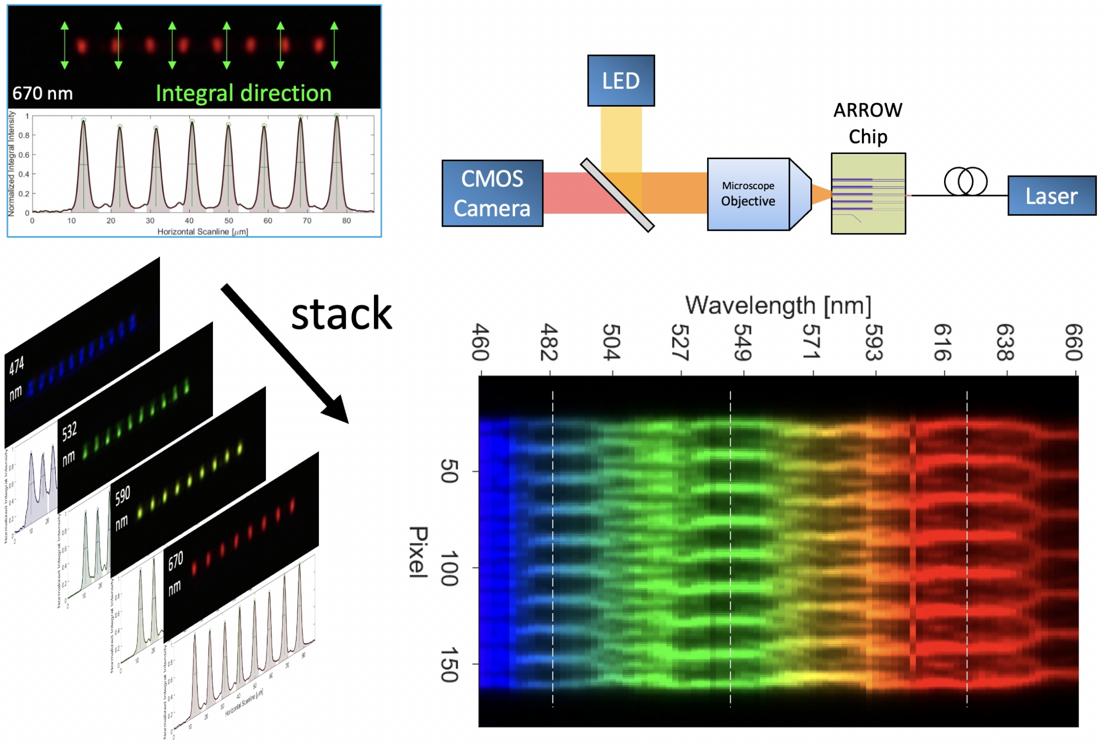
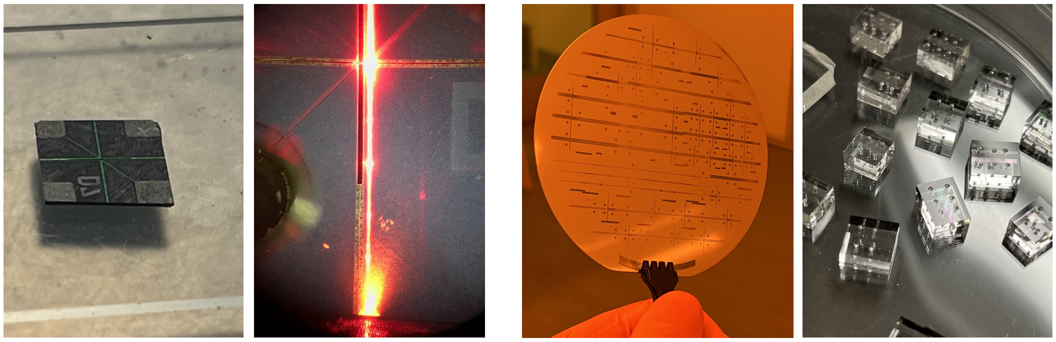


Figure 1.8: Automated MMI pattern characterization setup.

software converts image frames into 1D arrays by summing up the pixel intensity values inside a region of interest (ROI) box. This step repeats while the input wavelength is swept and by stacking those 1D plots, we get a full 2D map of MSE patterns versus the wavelength. By looking at this 2D map, we can determine the right wavelengths for integer N spots, as well as study how these patterns can vary from chip to chip.

Optical waveguides and microfluidic channels can be made using other materials, such as polymers. Polydimethylsiloxane (PDMS) is one of the popular materials for biosensing applications due to numerous factors such as ease of prototyping and fabrication, high transparency in the visible spectrum, tunable optical properties, and



ARROW chip

ARROW chip in the setup

SU-8 master for PDMS chip fabrication

PDMS chips

Figure 1.9: SiO_2 (left) and PDMS (right) MMI biosensors.

price and availability. Our optofluidic biosensors have been successfully used in several studies [8, 9, 10]. Fig. 1.9 shows MMI biosensor chips fabricated with silicon dioxide standard fabrication process (left) and PDMS soft-lithography fabrication process (right). The soft-lithography fabrication process requires one (or more) masters with features created using SU-8 photoresist which then will be used to cast the features (waveguides, channels, etc) into the PDMS layers. Check [8, 11, 12] for more detail in PDMS-based optical and optofluidic device fabrication.

After this introduction and background about optofluidic biosensors, we realize that most previous research and studies have evolved the optofluidic biosensor platform to an outstanding performance by optimizing hardware and device-level optimizations. In this thesis, the focus is mostly on data analysis to get more from currently optimized devices and is outlined as follows. Chapter 2 presents a study done on 3D hydrodynamic focusing implementation on conventional ARROW optofluidic design. It covers chip characterization with different practices and presents the analysis results. In Chapter 3, a novel, high-performance signal analysis technique based on continuous wavelet transform (CWT) is presented and includes comparison results with previously established methods. It also covers various sensory data types, such as fluorescence signals from optofluidic chips, electrical signals from nano-pore devices, and simulated protein mass spectra to investigate the broad application for the proposed robust event/pattern detection algorithm. Chapter 4 is about machine learning and implementing neural network-based classifiers for biosensors application. The benefits and enhancement achieved by the neural network are compared with previous methods and a demonstration of a real-time event detection framework on the edge is discussed at the end. Finally, In Chapter 6 an adaptive closed-loop system is designed and implemented to extend the detection dynamic range for ARROW biosensors to cover a range of concentrations while maintaining multiplexing capability. Appendix A provides more detail on the simulation model created for multi-factor statistical analysis of the fluorescent events from ARROW devices.

Chapter 2

3D Hydrodynamic focusing in optofluidic devices

This chapter presents an ARROW biosensor device with three-dimensional hydrodynamic focusing (3DHDF) functionality. It covers the pros and cons of adding the 3DHDF feature into an optofluidic design and how we tried to optimize this concept for point-of-care applications. Three different designs are explored with practical challenges, steps taken to solve those challenges and characterization results with fluorescence measurement. We emphasize that this study is done in collaboration with Brigham Young University (BYU). Chip design, finite element simulation, and fabrication are done in Prof. Aaron Hawkins' group at BYU and chip characterization via fluorescence imaging and fluorescent bead time trace analysis is done in Prof. Holger Schmidt's lab at UCSC.

2.0.1 Hydrodynamic focusing in a microchannel

The microfluidic channel in our ARROW biosensors is a micron-scale hollow-core channel to guide light while flowing the analyte. The liquid inside the ARROW chip flows in a laminar regime because of a small Reynold's number *Re*. The *Re* for a circular channel is described by

$$Re = \frac{\text{inertial forces}}{\text{viscous forces}} = \frac{\rho \bar{v} D}{\mu} \tag{2.1}$$

where ρ is the density of the fluid, $\bar{v}$ is the average velocity of the fluid, D is the diameter of the channel, and μ is the viscosity of the fluid. For a rectangular channel (duct) we can write the equation by using the hydraulic diameter (D_h) determined as

$$D_h = \frac{4A}{P} \tag{2.2}$$

where A and P are the cross-section area and perimeters of the rectangular duct respectively. The *Re* hence is defined by

$$Re_{D_h} = \frac{\rho \bar{v} D_h}{\mu} \tag{2.3}$$

considering the small dimensions of ARROW biosensors, the microfluidic channels are typically in microns resulting in $Re < 2000$, therefore the flow of the sample inside the liquid channel is laminar and forms a parabolic velocity distribution. This laminar flow means that particles flowing near the wall have the minimum speed (zero for non-slip wall conditions), and those flowing at the center of the cross-section have the maximum speed. In terms of particle detection, stretched or compressed events in the time domain are observed in the fluorescence signal. Excitation spots within the

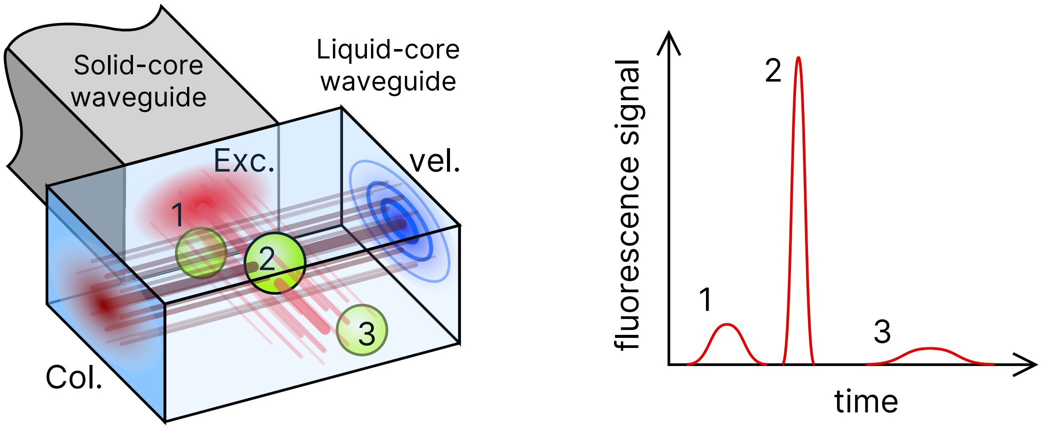


Figure 2.1: Fluorescence signal variations as a result of non-uniform excitation, collecting, and velocity profiles inside the excitation volume.

channel follow the mode profile of input light, which for a single-mode input waveguide resembles the distribution

$$
\begin{aligned}
E(x) &= \cos(k_x x) \text{ in core} \\
&= \begin{cases} e^{-\gamma x} & \text{in cladding for regular WG} \\ \text{oscillation} & \text{for ARROW} \end{cases}
\end{aligned}
\tag{2.4}
$$

Ideally, the brightest part of excitation is located at the center of the channel and the intensity tapers off towards the edges. Moreover, the collection efficiency of fluorescence inside a liquid-core waveguide depends on the particle location [13, 14]. Overall, detecting particles flowing close to the edges becomes a real challenge considering these distribution-related weakened signals. Fig. 2.1 visualizes these non-uniform distributions inside an optofluidic excitation volume and expected fluorescence signals of particles flowing at different locations of the analyte channel. Two-dimensional hydrodynamic focusing has already been shown to improve performance [15, 16] which encourages the potential for further improvement by adding the vertical HDF into the design. 3DHDF, which centers target particles in the cross-section of a fluid channel, is considered to significantly improve the detection efficiency in a planar chip-scale biosensor.

2.1 3DHDF designs for ARROW optofluidic devices

Hydrodynamic focusing can be accomplished using buffer fluid to sheath sample fluid in a microscale channel. When the buffer fluid flows are supplied from the sides of the sample fluid a two-dimensional focusing is formed and the sample stream is horizontally compressed [15]. Three-dimensional focusing, in which the sample stream is surrounded by buffer streams laterally and vertically, is more challenging [17] particularly, in microfluidic devices with planar architecture as delivering buffer fluid from above and beneath the sample fluid channel adds complexity and can require multiple vertically stacked layers. This complexity in fabrication and even in operation, sometimes demands additional fluid ports and channels. Our goal for creating a 3DHDF microchannel design is to minimize the complexity of this concept and implement 3DHDF in a chip-scale optofluidic biosensor. Several 3DHDF schemes have been proposed that utilize interesting flow phenomena but unfortunately, these designs are unusable when cross-section geometries approach the 10-μm range [18, 19, 20, 21]. Methods that rely on the secondary-flow vortexes become impractical due to the very high fluid velocities and enormous fluidic resistance caused by the small cross-section of the microscale

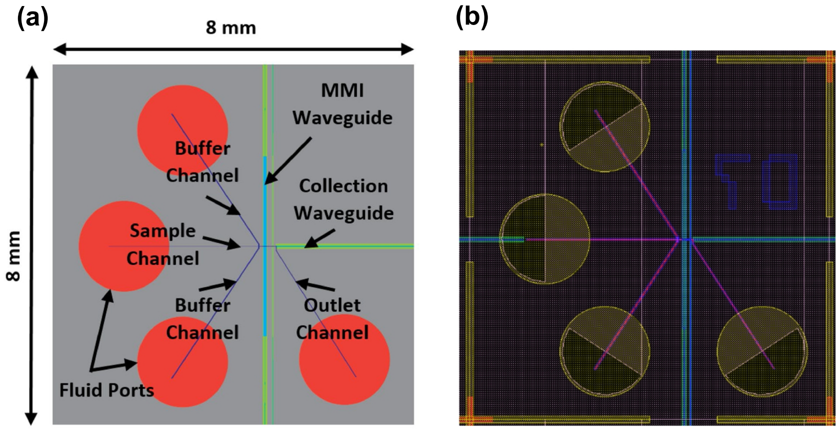


Figure 2.2: Trench chip layout (a) import elements of the chip (b) mask layout.

channels [16]. To address this issue, a low-velocity scheme is required, in which high flow rates are not required to achieve the desired focusing effect. Here we present a 3DHDF design based on sheath flow that can achieve focusing over a range of fluid velocities, including low velocities. This design was inspired by earlier demonstrations at larger scales [22, 23, 24]. The previous designs were complicated and require multiple precise fabrication steps. Also due to multiple sheath flow channels, several fluid ports (six) were needed to operate the chip. Here, we explore a scheme requiring only four fluid ports. The design is called trench design wherein a 3DHDF effect is formed via enclosing the main sample stream by lateral as well as top and bottom buffer flows. The trench and nub elements fabricated by two photoresist layers aid with supplying buffer flows in the vertical dimension.

2.1.1 Integrated 3DHDF Design

The illustration shown in Fig. 2.2 outlines the general design for our optofluidic chip that incorporates a 3DHDF element. The $8 \times 8mm^2$ silicon-based chip shown in Fig. 2.2a features four fluid ports: one sample port, two buffer ports, and an outlet port. The ports handle the fluid introduced into the microchannels, flowing from left to right. Optical waveguides orthogonally intersect the microchannel liquid-core waveguide to deliver excitation light and collect emitted photons from the target fluorescent particles. The are several upgrades in this layout compared to the original ARROW design. Simplification of the fluidic arrangement in this design reduces the chip footprint and yields a 45% increase in the density of devices per 100 mm wafer (from 52 to 76). The other upgrades are the extended waveguide design for the collection waveguide (see Fig. 2.2b) as well as marked cutting lines to ease and increase tolerance of the chip cleaving process. The extended collection waveguide from the neighboring chip is ending at the sample fluid port. Also, the circular fluid port pads compared to the square pads in the previous designs, match the circular profiles of fluidic interfaces, thus better sealing for the fluidic interface. The MMI excitation waveguide is located 100 μm from the trench part (fluid focusing junction) of the device.

The main part of the trench design is the prismatic plus-shaped fluid junction (Fig. 2.3) where the 3D hydrodynamic focusing effect happens. Fig. 2.3a shows the direction of fluidic flow into and out of the trench intersection. The trench and the nub sections of the intersection make sure that the sample stream is supported by a buffer

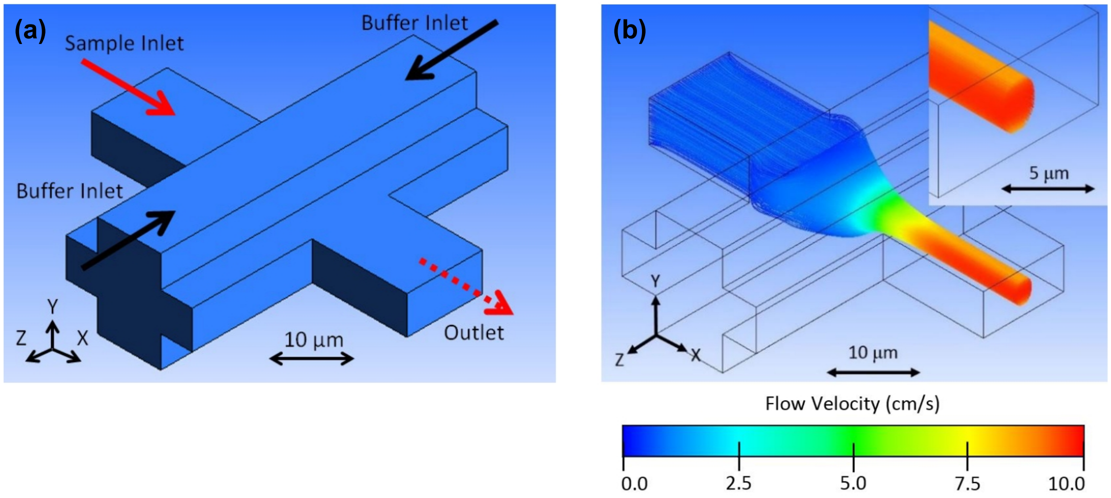


Figure 2.3: The prismatic plus-shaped junction of the trench design (a) schematic of the trench part with inlets and outlet ports (b) the CFD simulation of trench design predicting a nice symmetric 3D focusing.

Feature	Height [μm]	Width [μm]	Length [μm]	Velocity [cm/s]
Sample inlet	6	6.76	2,940	1
Buffer inlet	6	20	2,930	1
Trench/Nub	5	10	104	-
Outlet	6	12	3,285	-
Chip edge	-	8,000	8,000	-

Table 2.1: Trench design parameters.

in lateral and vertical directions, hence delivering horizontal and vertical focusing. Fig. 2.3b displays the 3D focusing behavior of the trench design simulated by a commercial computational fluid dynamics (CFD) package called ANSYS Fluent (done by BYU). The prismatic plus-shaped offers a nice round focused stream inside the rectangular outlet channel.

Table 2.1 summarizes the design parameters extracted from the CFD simulations. Note that the sample and buffer fluids' velocities are equal.

2.1.1.1 Fluorescence imaging

In order to explore the hydrodynamic focusing behavior of the chip fabricated at BYU, at UCSC a novel approach was developed to enable a 3D measurement by using only a single 2D top-down imaging setup. This is necessary due to the planar design of the chip which limits access to the sides of the chip for imaging needs. In this technique, the chip is driven in the inverted format of the original design, meaning that we swap the buffer and the sample channel fluids. The simplified cartoon in Fig. 2.4a depicts the top-down fluorescence microscope imaging configuration exploited to characterize 3DHDF behavior. When the device is filled in the original mode, the sample channel is filled with a fluorescent dye (here, 10 μM Cy5 dye) and buffer channels are filled with buffer (here, DI water). All three possible cases, no focusing, 2D focusing, and 3D focusing are visualized in Fig. 2.4b using two different projection planes (side, and top) to help understand expected fluorescence images from the chip. When there is no focusing behavior in an optofluidic chip (i.e. when both sample and buffer channels are filled with fluorescent dyes), a bell-shaped curve is predicted for the fluorescence signal. The fluorescence signal is the average of pixel intensities from the top-down image along the z-axis. Notice the similarity of all three scenarios with the original scheme of operating the 3DHDF chip where all fluorescence signals render a bell-shaped profile.

The original scheme would not let us distinguish between 2D and 3D focusing behaviors, so in order to tackle this problem, we devised an unconventional format for

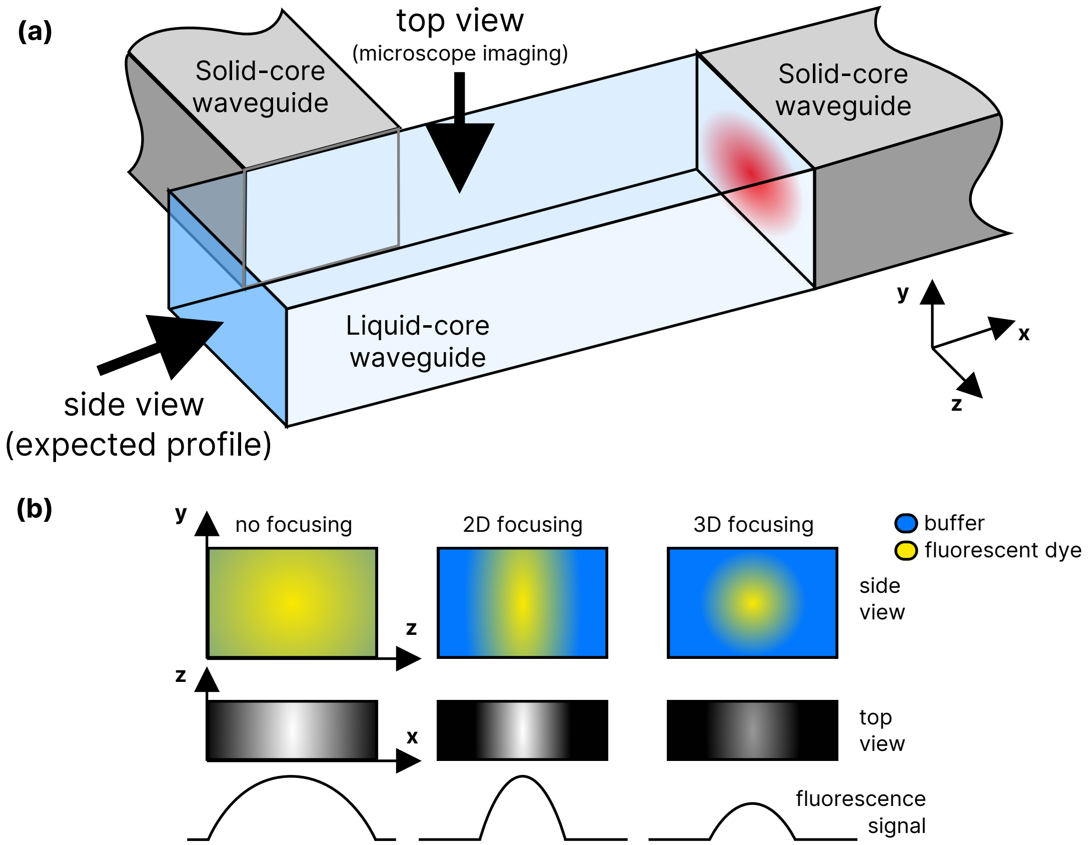


Figure 2.4: Predicted fluorescence signals in original format (a) schematic of excitation-liquid channel intersection. Unlike the usual setup, here we shine light from the collection waveguide to illuminate the full length of the liquid-core channel. (b) predicted fluorescence images and intensity signals for three possible cases. The top-down microscope imaging produces very similar intensity profiles for 2D and 3D focusing cases.

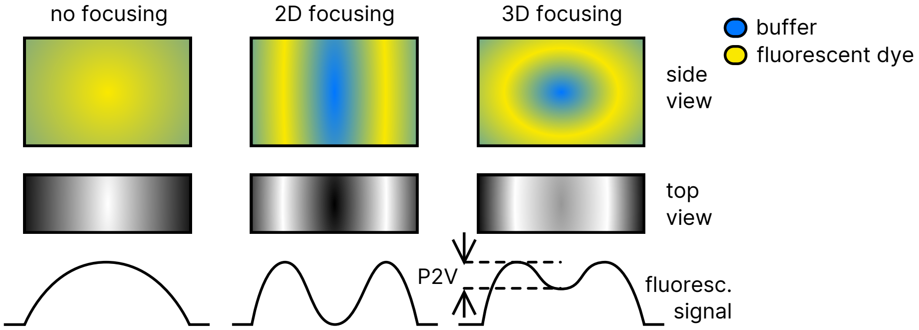


Figure 2.5: Predicted fluorescence signals in inverted format. The intensity signal is different for each case making it possible to distinguish three possible behaviors by just single top-down imaging. Peak-to-valley (P2V) is a smaller value in 3D focusing compared to 2D focusing due to dye presence on the top and bottom of the focusing stream.

driving HDF chips to introduce the contrast within fluorescence images. In this scheme, we introduce fluorescent dyes into the buffer inlets and fill the sample channel with buffer. This inverted scheme is expected to produce profiles shown in Fig. 2.5 by introducing a new peak-to-valley (P2V) feature in the fluorescence signal. A narrow stream of dye on the top and bottom of the 3D-focused stream raises the valley decreasing the P2V value in comparison to the 2D-focusing scenario.

2.1.1.2 3D focusing characterization

In an experiment, we use the inverted scheme to evaluate the hydrodynamic focusing behavior of a trench chip. The buffer channels are filled with Cy5 fluorescent dyes with a concentration of 10 μM and excited by a side coupled 633 nm laser source (NKT SelectBox) exciting the entire analyte channel from the collection waveguide. We choose a collection waveguide instead of multimode interference (MMI) waveguide so that the full length of the analyte channel gets illuminated. Fluorescence images are taken using a compound fluorescence microscope with a 700 nm long-pass optical filter to cut off the excitation light. A 20× microscope objective and an sCMOS camera (Andor Zyla) capture the top-down images. The image shown in Fig. 2.6 is an example of a top-down fluorescence image taken from a trench 3DHDF chip in the inverted scheme and the narrowly focused stream of DI water (darker than the surrounding bright Cy5 stream) is visible. A dim backlight was used to illuminate the main parts of the chip like channels and solid-core optical waveguides. Fig. 2.6 shows the intensity profile of the fluorescence image with a visible valley proving the 3DHDF effect.

$P2V_{\text{normalized}}$ described by equation 2.5 is used to plot the P2V changes along the x-axis.

$$P2V_{\text{normalized}} = \frac{P2V}{\text{Max}(I) - \text{Min}(I)} \tag{2.5}$$

Where $P2V$ is the peak-to-valley measured from the intensity curve (I) of a fluorescence image within an averaging box (see Fig. 2.7a). The analysis was done with the inverted scheme and averaging of pixel values along the x-axis within an averaging box of 30 μm for Fig. 2.7b and 5 μm for Fig. 2.7c. By analyzing the $P2V$ metric along the analyte channel, we realized that the 3DHDF effect can only last for a limited distance (<150 μm) and is maximized at a distance 20-30 μm from the trench junction.

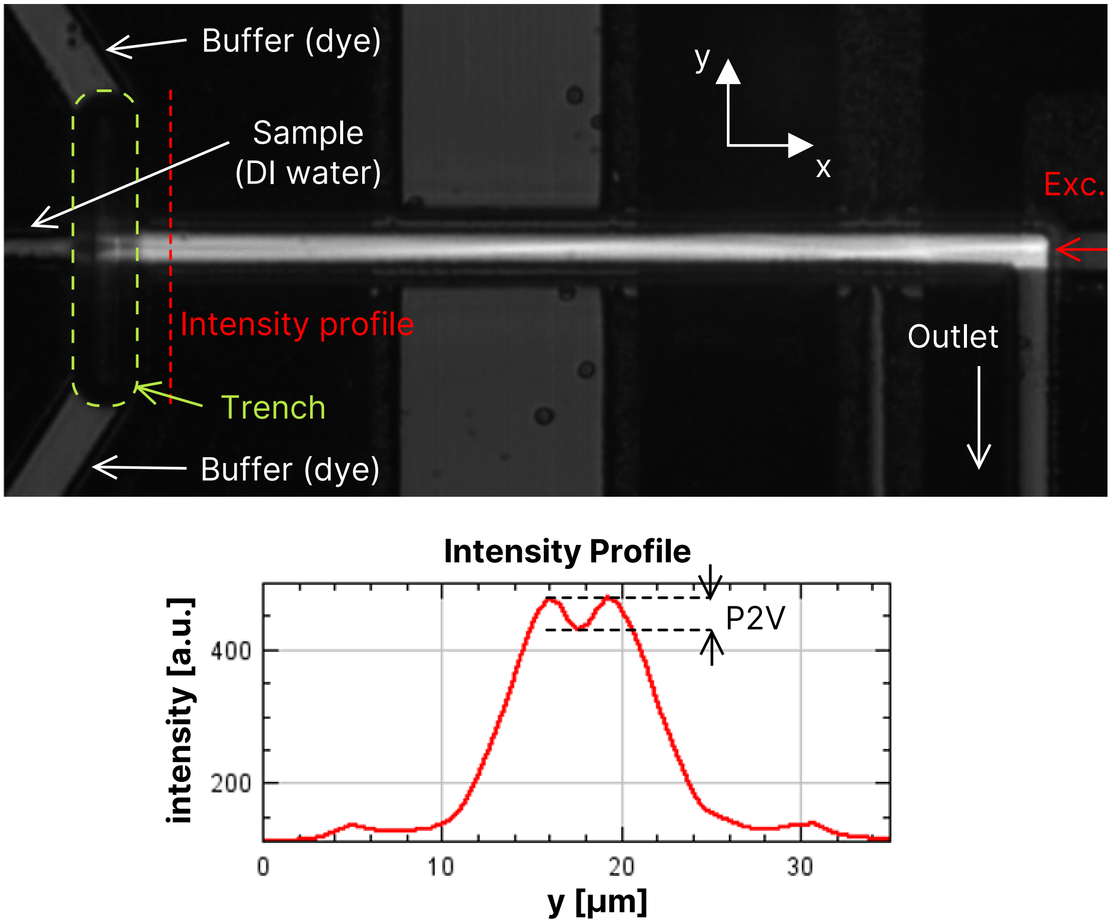


Figure 2.6: Experimental analysis of 3DHDF behavior using inverted scheme. The predicted intensity profile is observed from the fabricated chip with a small P2V.

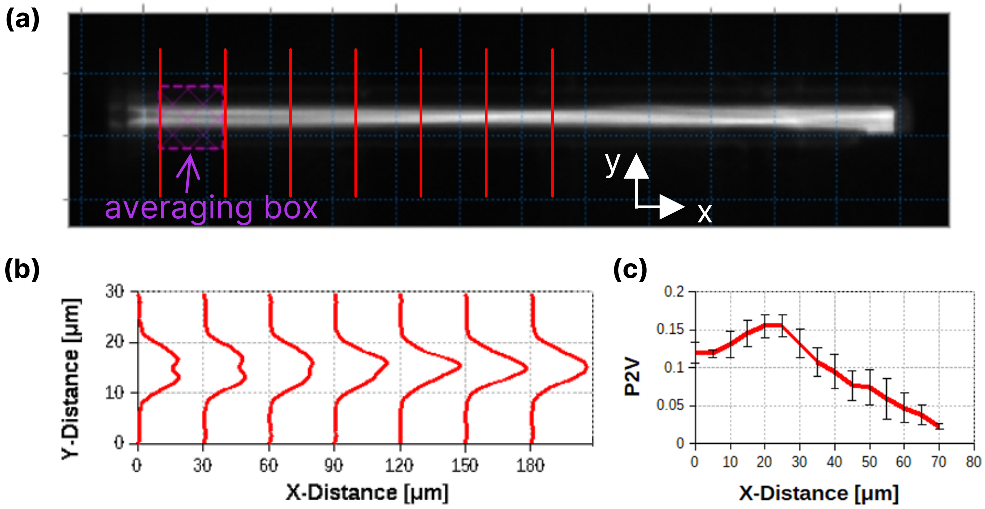


Figure 2.7: P2V analysis along the analyte channel in the inverted scheme.

2.1.1.3 Diffusion characterization

The fading effect observed in Fig. 2.7 is related to hydrodynamic diffusion and is studied in this section. By switching back to the normal mode of operation for the trench 3DHDF chip, we analyzed the 3DHDF effect by analyzing the MMI pattern formed at the excitation volume. The input light is 633 nm and is coupled into the collection waveguide similar to the previous setup. A narrow stream of dye focused in the center of the channel is shown in Fig. 2.8. This narrow stream gets diffused and wider as it flows along the channels and a profile analysis is employed by dividing the length of the analyte channel into several slices (see Fig. 2.9). We derived equation 2.6 to predict the FWHM by first calculating a concentration, c, as a function of lateral y position and microchannel length position x (see Fig. 2.9 for x and y axes). The model is simplified for a 1D semi-infinite diffusion problem with a stream half-width a.

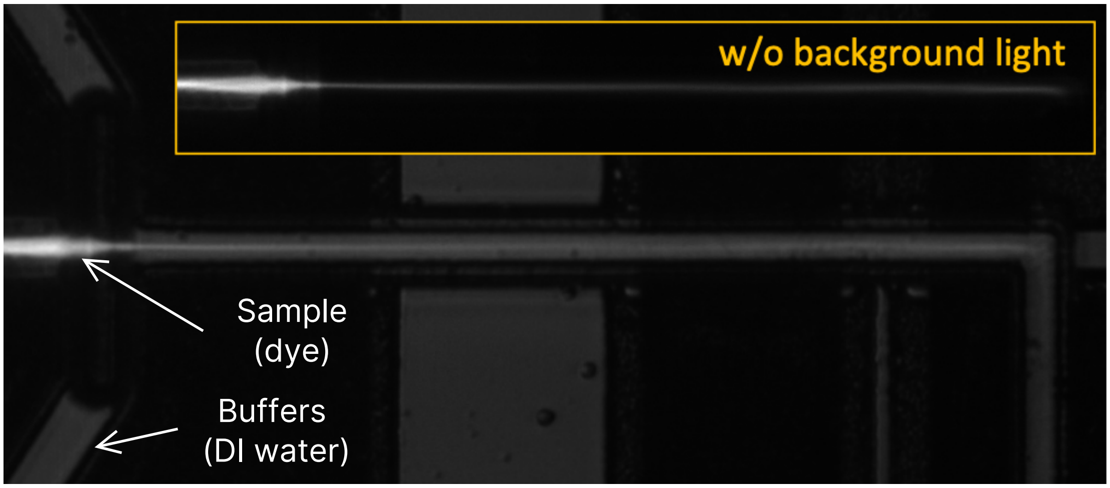


Figure 2.8: 3DHDF chip operating in normal mode. The sample channel is filled with dye and the buffers are filled with DI water. A narrowly focused stream of dye is noticeable, especially in the inset without any backlight.

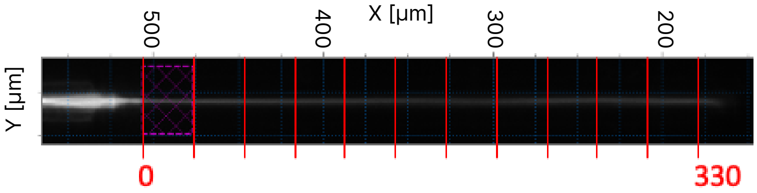


Figure 2.9: Focused stream starts to diffuse along the analyte channel.

κ represents diffusivity divided by flow velocity [25].

$$c(x,y) = \frac{1}{2}\left\{erf\left(\frac{a-y}{\sqrt{4\kappa x}}\right) + erf\left(\frac{a+y}{\sqrt{4\kappa x}}\right)\right\}. \tag{2.6}$$

Equation 2.5 is used to calculate the predicted FWHM shown in Fig. 2.10 to compare with the FWHM values extracted from the experiment. Measured values are those drawn out from Fig. 2.9. The predicted FWHM for 1 kilobase pair single-stranded DNA is of interest in our ARROW biosensors for the detection of biological samples, i.e.

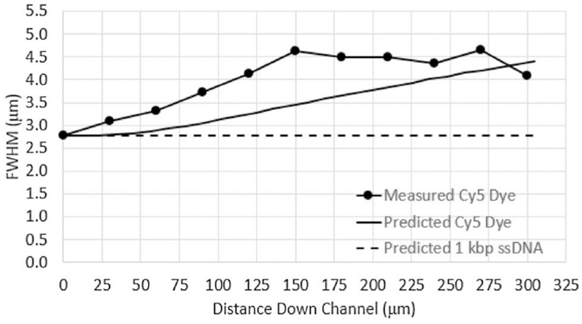


Figure 2.10: Diffusion of different targets within the 3DHDF channel.

antibiotic-resistant bacteria. The prediction curve for 1 kbp ssDNA shows a negligible diffusion which is much lower than the diffusion predicted for Cy5 dye [26, 27]

The trench 3DHDF chip incorporates the MMI excitation design and this was used to further characterize the focusing behavior. This time, we filled the sample channel with a fluorescent dye and buffer channels with DI water just like in the diffusion study. By varying applied negative pressure, the pressure difference between inlet channels and outlet channels is varied and the 3DHDF effect shows a pressure-dependent focusing behavior. Fig. 2.11 shows a few examples of fluorescence images taken from the chip driving fluidic sample at four pressures (10 inHg, 15 inHg, 20 inHg, and 35 inHg).

Obviously, by pulling the sample with stronger vacuum pressure, the focused stream becomes narrower at the excitation volume. The chip was excited by a He-Ne laser running at 633 nm which produces the expected 7 peaks inside the channel. We take advantage of this spatial information from the wide MMI excitation pattern to accurately construct a pressure-dependent focusing curve. The excitation length,

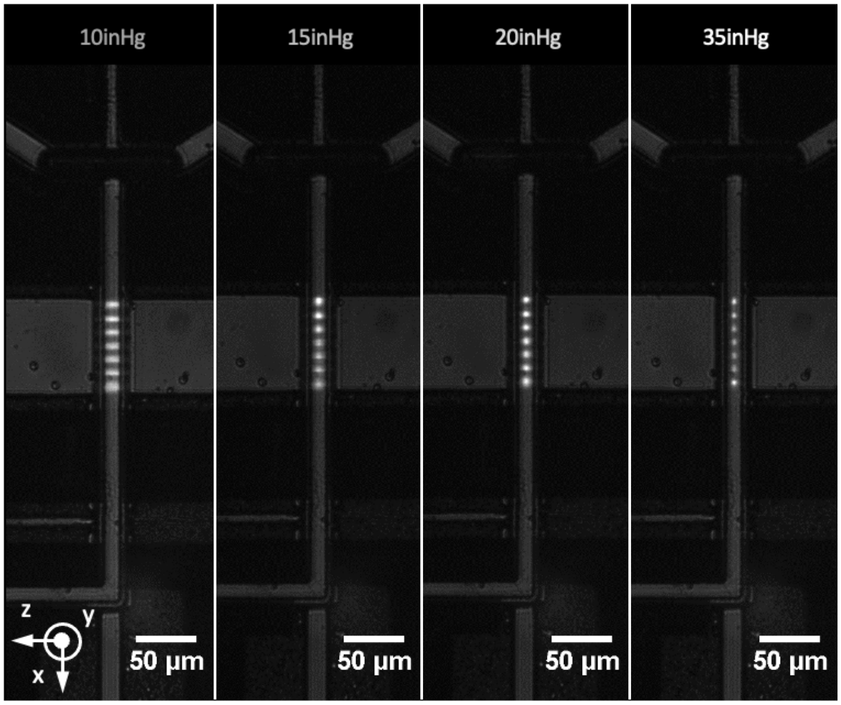


Figure 2.11: 3DHDF effect on MMI pattern over vacuum pressures range. 3DHDF effect is stronger at stronger vacuum pressures.

defined as the average FWHM of all 7 spots, versus applied pressure, is plotted in Fig. 2.12. Even a small divergence of the focused stream is visible within 75 μm wide MMI excitation volume (see Fig. 2.12 left). A 50% narrowing happens at a pressure range of 15-25 inHg and this narrowing effect goes over 65% at 35 inHg.

If we individually analyze the FWHM changes of each spot versus applied pressure, we see a decrease in the slope (reflecting the drawn envelope in Fig. 2.12 left). In other words, the MMI spots become more uniform as we improve the focusing effect by applying stronger vacuum pressure (see Fig. 2.13). This unique spatial information attained by multi-spot patterns can be helpful for diffusion-based detection methods, especially at slower flow rates.

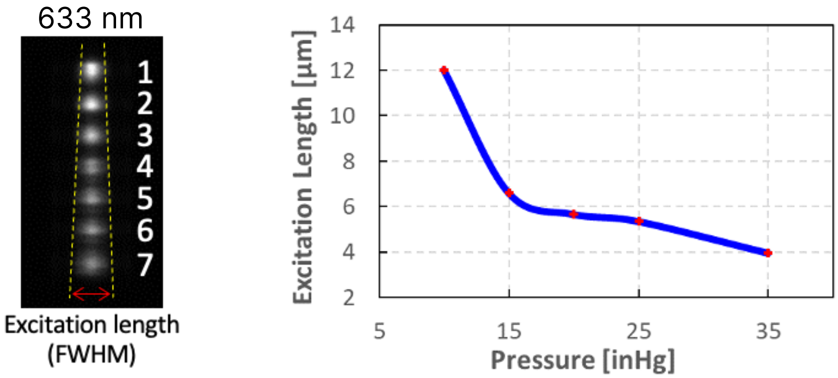


Figure 2.12: Excitation length versus applied pressure. The excitation length is the average of FWHM of 7 peaks for 633 nm excitation.

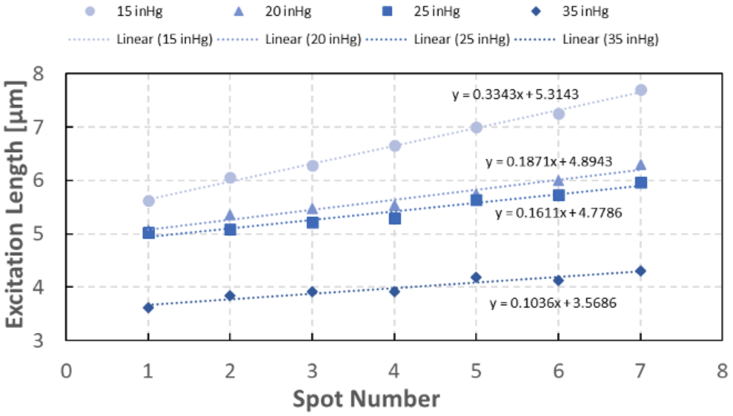


Figure 2.13: Individual spots' width versus applied pressure. A similar narrowing trend is observed for all individual MMI spots. At stronger vacuum pressures, the MMI pattern is more uniform and the divergence of spots is minimal.

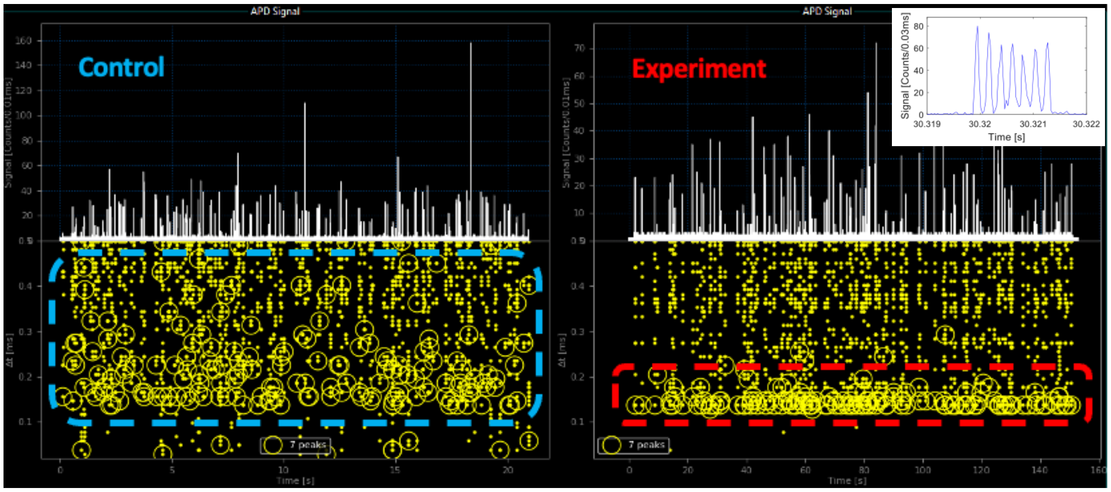


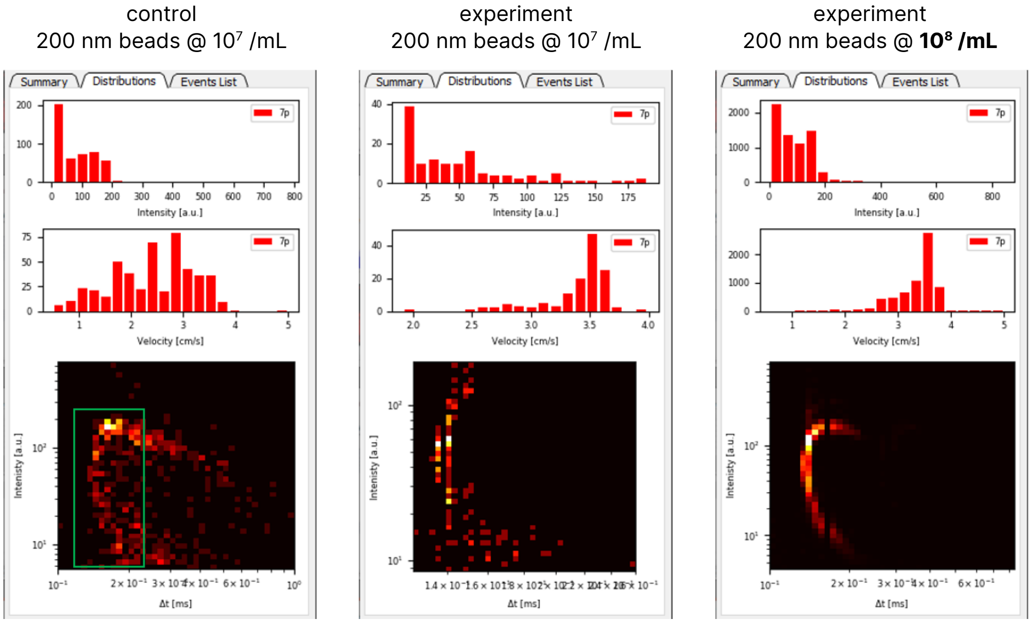
Figure 2.14: APD traces of control (no focusing) and experiment (3D focusing) of 200 nm crimson fluorescent beads. Detected events are distributed in a much narrower Δt range at the lower Δt axis. Yellow rings indicate detected events in time-Δt space using CWT analysis.

2.1.1.4 Fluorescent particle detection in 3DHDF chip

Finally, the trench design was validated by applying a solution containing 200 nm fluorescent beads to investigate the digital detection of nanoscale targets. Fig. 2.14 shows APD traces recorded for the sample with 200 nm crimson beads (FluoSpheres$^{\mathrm{TM}}$ Carboxylate-Modified Microspheres, Invitrogen$^{\mathrm{TM}}$) with a diluted concentration of 10^7 /mL. The control is when all channels are filled by the sample and the experiment is when buffers are filled with DI water. The inset shows an example event representing 7 distinct peaks from the multi-spot excitation pattern. Events are detected using the developed PCWA algorithm [28].

Statistical analysis of the detected events reveals a narrowing in fluorescence signal intensity as well as velocity. The signal intensity for an event is the CWT coefficient of that detected event. The correlation of fluorescent particle intensity with its flowing speed can be visualized by plotting a 2D join-histogram as shown in Fig. 2.15. The column on the left is for the APD trace recorded from the control experiment, where all channels are filled with dye. As expected, the velocity and intensity histograms are wide in this case and a broad horseshoe shape is observed in the 2D histogram. The middle column depicts plots for the experiment trace with narrower histograms. Also, we tried a 10× higher concentration of the sample (10^8 /mL) to get a higher count of events for histogram comparison and saw very similar distributions to the 10^7 /mL trace (see Fig. 2.15 right column). Notice the change in the horseshoe shape in the 2D histogram with shorter legs, especially in the Δt/velocity axis. Comparing the control and the experiment (10^8 /mL) analysis, we see a significant shift and compression in the velocity distribution, but a minor shift for the average intensity. The increase in mean velocity is related to the reduced viscosity of the sample pulled through the chip since the buffer channels are filled by water instead of beads. The average intensity is reduced in the experiment which is because of misalignments from the fabrication. The optimal alignment should center the three main elements, the excitation waveguide, the analyte channel, and the collection waveguide. This plays a significant role in APD trace signals for 3DHDF chips since there is only a very narrow stream of targets crossing the excitation volume.

The simulation model developed in Appendix A is used here to seek the effect of various focusing on 2D joint-histogram shapes. Fig. 2.16 shows a simulated device by adding excitation and collection profile offsets as:


control
200 nm beads @ 10^7 /mL

experiment
200 nm beads @ 10^7 /mL

experiment
200 nm beads @ $\mathbf{10^8}$ **/mL**

Figure 2.15: Velocity and signal intensity distributions as well as 2D joint-histogram for control and experiment. The 10^8 sample produces a higher number of events/second, thus a better histogram is formed. The 3D focusing effect impacts velocity distribution more than the intensity distribution.

- $\mu_{ex,y} = 3\mu m$
- $\sigma_{ex,y} = 0.2 \times h = 1.2\mu m$
- $\mu_{c,x} = 0\mu m$
- $\mu_{c,y} = -4\mu m$
- $\sigma_{c,x} = 0.5 \times w = 6\mu m$
- $\sigma_{c,y} = 0.33 \times h = 2\mu m$

We simplify the focusing effect by just masking (cropping) the velocity profile in three different modes: laterally and vertically for 2D focusing, and both for 3D effect (see upper left insets in Fig. 2.14). The second, third, and fourth columns of the top row insets represent excitation, collection, and sensitivity profiles respectively. As you can see in the right 2D joint-histogram (3D focusing), the produced histogram reflects a similar transformation observed with the experimental data, which confirms the 3DHDF is indeed at work in the 200 nm fluorescent bead experiment.

2.2 Operating 3DHDF chip with 3D printed manifold

The microfluidic part of a typical ARROW biosensor relies on only a single fluidic path from one inlet to one outlet. While this simple microfluidic design works well with conventional single-line fluidic interfacing, it becomes impractical to operate 3DHDF devices with multiple inlet/outlet ports. Fig. 2.17 shows a conventional fluidic interface for an ARROW chip with a vacuum-driven sealing of the chip reservoir and

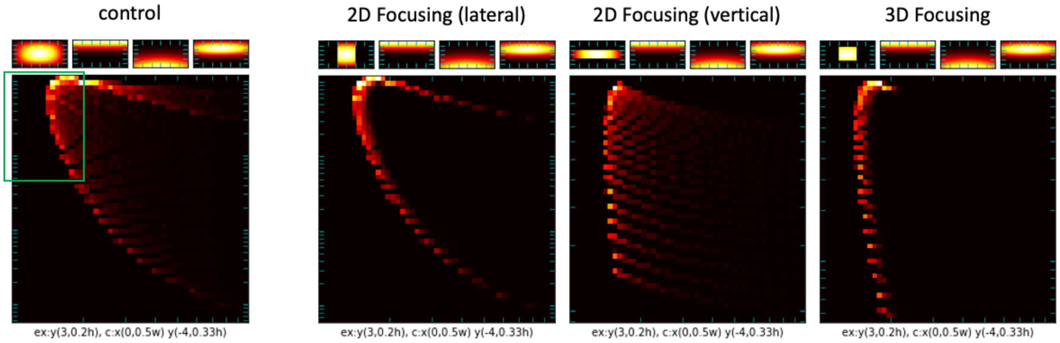


Figure 2.16: Velocity and signal intensity distributions for control and experiment. The narrowing effect is observed for both distributions in the experiment case.

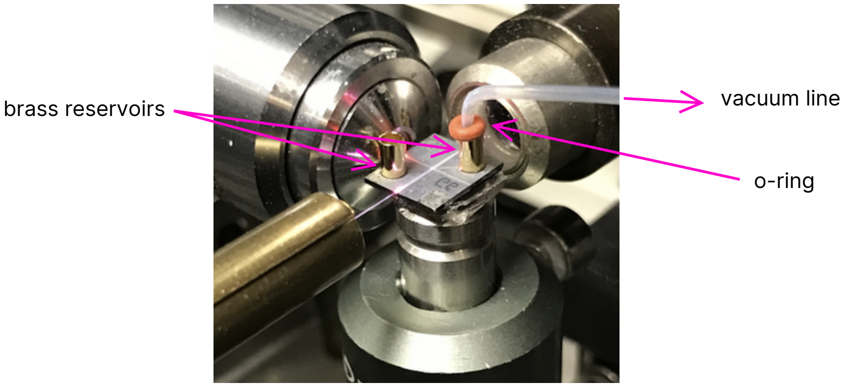


Figure 2.17: Conventional ARROW fluidic interface.

outlet vacuum line. The o-ring (in orange) helps with providing air-tight sealing to avoid pressure drops in the reservoir.

The 3DHDF devices usually host multiple inlets and outlets to aid required microfluidic streams running in the chip. The small footprint of these chips makes the fluidic interfacing challenging, thus demanding a new and different mechanism. Fig. 2.18 (top-right) shows an example of a 3DHDF ARROW chip (stacked channel design [23]) with three inlets and three outlets. There are two sealing o-rings placed on top of two outlets to demonstrate how tight space between reservoirs makes it difficult to connect all outlets into dedicated fluidic handling lines. A new part was designed and 3D-printed using high-resolution stereolithography (SLA) printers in-house (UCSC BELS). The CAD file was created in Autodesk Inventor software and is shown in Fig. 2.18. The 3D printed part incorporates two separate parts. The bottom part is to attach to a post mounted on an X-Y stage (see Fig. 2.17) and hold the chip in place via a suction mechanism (see Fig. 2.18 top-left). The top part provides access to the inlets and outlets via a manifold design. The connecting ends of manifolds include o-rings for tighter sealing into the chip without any need for brass reservoir attachment. The other end of the manifolds (the vacuum line) has a tapered insert design to provide tight fitting to the tubings. In a revised version, we extended the manifold part to offer bigger clearance over the chip for top-down imaging purposes.

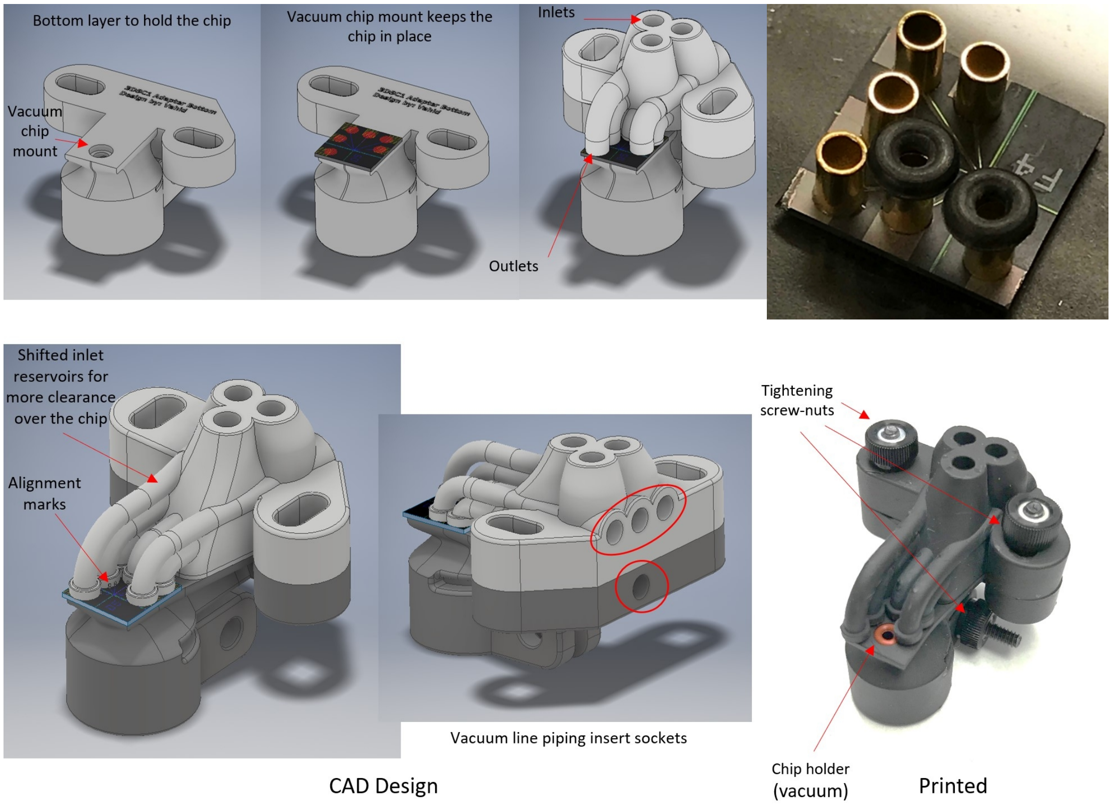

Bottom layer to hold the chip
Vacuum chip mount
Vacuum chip mount keeps the chip in place
Inlets
Outlets
F4
Shifted inlet reservoirs for more clearance over the chip
Alignment marks
Vacuum line piping insert sockets
Tightening screw-nuts
Chip holder (vacuum)
CAD Design
Printed

Figure 2.18: Custom-designed 3D printed chip mount for 3DHDF chips provides sealed access to 6 inlet/outlet channels. Autodesk's Inventor software was used to draft the 3D model and the part was printed using the Formlabs SLA 3D printer available at UCSC BELS. The bottom row shows the revised design with a bigger clearance for the top-down imaging microscope objective.

Chapter 3

High-performance signal processing technique for particle counting

Chip-scale diagnostic devices as the current trend toward affordable and portable point-of-care (POC) platforms are increasingly incorporating microfluidics in their design [29, 30]. Microfluidics answers the need for portability, small sample volume, short processing time, integration, and flexibility. The integration of photonics and microfluidics brings biological detection into a chip-scale format [31, 32]. ARROW devices as one of the successfully developed optofluidic biosensors have been demonstrated at clinically relevant concentration ranges with single nucleic acid, proteins, viral particles sensitivity and multiplexed detection capability [33, 34]. These devices rely on fluorescence detection of single particles flowing inside a liquid-core waveguide. Detection, identification, and counting of fluorescently tagged particles in a precise and efficient way is the subject of this chapter.

3.1 Fluorescence detection

In this chapter, we discuss a new high-performance event detection technique developed for ARROW fluorescence biosensors, but the application of this technique is not limited to these devices. A couple of examples where certain patterns (here peaks) are asked to be detected are analyzed using this technique to give the reader an impression of how they can apply and use this tool for their own needs. However, we have found that it is particularly attractive for a liquid-core waveguide optofluidic platform on which an intersecting solid- and hollow-core antiresonant reflecting optical waveguides (ARROW) design introduces some variation of signals requiring special signal processing techniques. The target particles flowing at different velocities across the liquid-core channel, alongside the excitation and the collection waveguides' non-uniform mode profiles, result in a complex variation of signal properties. Fig. 3.1 demonstrates a conceptual illustration of orthogonally intersecting solid- and liquid-core waveguides with a fluorescent particle getting excited while flowing inside the channel. Depending on the position of the particle, excitation, emission, collecting as well as the time duration of the signal will vary.

The ARROW chip and simplified experimental setup used for fluorescence detection are shown in Fig. 3.2a. The chip consists of a $5\mu m \times 12\mu m$ microchannel accessed by fluidic reservoirs to introduce target particles by applied pressure. An external laser source is coupled into a single-mode optical fiber to inject light into the chip via facet coupling. As shown, the chip layout features two excitation options: single-spot via a

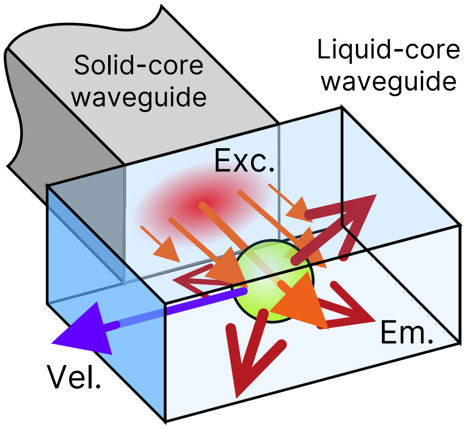


Figure 3.1: Cartoon of integrated optofluidic fluorescence detection. Excitation, fluorescence emission, and velocity vectors for a flowing fluorescent bead are illustrated

single-mode (SM) waveguide or multi-spot excitation (MSE) with a multi-mode interference (MMI) waveguide. The top-down microscopy image of the excitation region when the channel was filled with quantum dots in DI water, reflects two possible excitation regions colored in red and green (Fig. 3.2b). This high-concentration solution creates a static image of the excitation patterns through which a single target particle moves in an actual experiment. Single-spot excitation is useful when ultrasensitive singleplex detection is required as demonstrated by amplification-free detection of Ebola virus RNA [35]. MMI waveguides, on the other hand, create spectrally and spatially varying excitation patterns that have been successfully used for multiplexed single virus and antigen detection [10, 36, 34]. Figure 3.2c shows fluorescence signals examples corresponding to single-spot and multi-spot excitation generated by SM and MMI waveguides. They are recorded from fluorescent nanobeads and labeled bacterial DNAs (Klebsiella pneumoniae carbapenemase), respectively. Encoding the spectral information into spatial information by using the MMI waveguide, embodies additional information into indi-

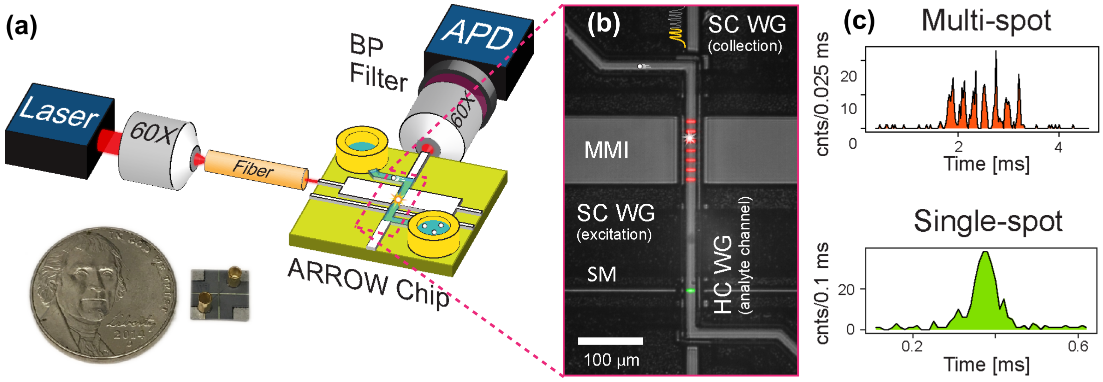


Figure 3.2: Optofluidic fluorescence detection platform. (a) Schematic of the experimental setup with orthogonal excitation detection paths. Single-peak or multi-peak modes are selected depending on which waveguide is coupled to the fiber. (b) Microscope image of detection region excited at 556 nm in the single-mode waveguide and 633 nm in the MMI waveguide which generates seven distinct spots in the analyte channel (channel filled with quantum dot-DI water solution for visualization of the excitation patterns). (c) Examples of generated fluorescence signals: Single-peak signal is from 200 nm diameter fluorescent nanobeads and the multi-peak signal is from a single fluorescently tagged nucleic acid from a Klebsiella pneumoniae carbapenemase (KPC) bacterium.

vidual events. When the multi-spot signal is analyzed by a shift-multiply algorithm (see below), a $50,000\times$ signal-to-noise (SNR) enhancement was demonstrated [37]. Detection and classification of these temporal events can be challenging when the background is high SNR is low as a result of a dim target or a particle's position in the channel [38]. Therefore, an efficient, powerful, and accurate signal analysis method is needed.

3.2 Wavelet transform

The velocity-dependent nature of fluorescence signals encourages us to explore time-frequency analysis techniques. Wavelet analysis is a widely used technique in time-frequency studies and has been successfully applied to a broad spectrum of applications, including but not limited to denoising, baseline removal, spike detection in noisy signals, and image compression [39, 40, 41, 42, 43, 44]. Continuous wavelet transform (CWT) and discrete wavelet transform (DWT) can be used in multi-scale peak and event detection. DWT decomposes a sampled signal into non-overlapping sub-bands of frequencies. DWT is generally fast and efficient but lacks a sufficiently high resolution of scale/frequency. Unlike DWT, CWT provides high scale/frequency resolution which is one of the key pieces of information used in time-frequency analysis. CWT uses a convolution operation to compare the signal $f(t)$ to a temporal pattern of finite duration $-$the mother wavelet $\psi(t)-$. The enhancement energy map in CWT transform helps to identify occurrences of the mother wavelet pattern within the signal. Eq. 3.1 describes the CWT transform

$$C(t,s) = \langle f, \psi_{t,s} \rangle = \int_{-\infty}^{+\infty} f(t') \frac{1}{\sqrt{s}} \psi^* \left(\frac{t - t'}{s} \right) dt' \tag{3.1}$$

where $s > 0$ is a scaling factor that stretches or compresses the wavelet bases in time. $C(t, s)$ is, therefore, the correlation of the real signal with a scaled and dilated basis function which can be visualized in a 2D map of regularity through coefficients. Local maxima in the 2D map indicate the presence of a particular pattern (similar to the mother wavelet) and the scaling function, extends the pattern search into a broad range of scales. This multi-scale pattern recognition behavior is well-suited for fluorescence particle detection where particles flowing at different velocities produce scaled patterns detectable in time-scale (t, s) space. It is, therefore, convenient to relabel the scaling parameter s as Δt to extract meaningful information for events that can, in turn, be easily converted into the velocity of flowing particles due to the direct correspondence with the known spatial excitation patterns. This is a key advantage over DWT, whose limited number of scale levels is not sufficient to extract the continuous distribution of the velocity of flowing particles. We note that the convolution calculation at each scale can be done independently for CWT whereas in DWT higher-level coefficients depend on the lower level's values, which is compatible with parallel processing schemes. Indeed, the technique has been used in numerous applications such as mass spectroscopy [45], powder x-ray diffraction [39], seizure detection from EEG signal [40, 41], radar target detection [42] and trend detection, and estimation in hydrology and climate research [43]. However, a major challenge lies in dealing with large amounts of data in a fast and memory-efficient way as well as with more complex signal shapes such as the multi-peak signal of Fig. 3.2c. Time-frequency resolution of wavelets changes based on the scale, whereas in short-time Fourier transform (STFT or windowed Fourier) it is fixed. The wavelet $\psi_{u,s}$ has a time support proportional to s and a frequency support proportional to $1/s$. Fig. 3.3. shows two wavelets $\psi_{u,s}$ and ψ_{u_0,s_0} with proportional time and frequency supports. When s changes depending on the position of the box, the time and frequency width of the Heisenberg box vary but its area remains constant. The figure is taken from [46] with permission.

$$s\sigma_t \times \frac{\sigma_\omega}{s} = s_0\sigma_t \times \frac{\sigma_\omega}{s_0} = \sigma_t\sigma_\omega \tag{3.2}$$

This emphasizes that events detected at higher frequencies/velocities (smaller scales) have better time localization and poorer frequency localization, and slow-moving parti-

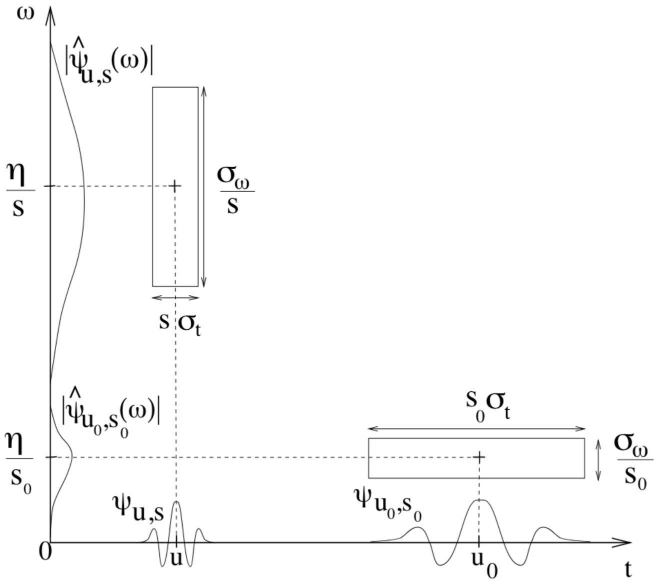

ω
$|\hat{\psi}_{u,s}(\omega)|$
$\frac{\eta}{s}$
$\frac{\sigma_\omega}{s}$
$s\,\sigma_t$
$|\hat{\psi}_{u_0,s_0}(\omega)|$
$s_0\sigma_t$
$\frac{\eta}{s_0}$
$\frac{\sigma_\omega}{s_0}$
$\psi_{u,s}$
ψ_{u_0,s_0}
0
u
u_0
t

Figure 3.3: Heisenberg uncertainty boxes.

cles show better frequency, but poor time localization resolution.

3.3 Event detection using wavelet transform

Figure 3.4a shows a fluorescence signal recorded with the optofluidic chip of Fig. 3.2 with the single-mode waveguide for excitation. A solid-state diode neodymium-doped yttrium aluminum garnet; Nd:Y3Al5O12 (SSD Nd:YAG) laser (Shanghai Dream Laser Technology Co.) coupled into the optical fiber, excites 0.1 pM of 200 nm polystyrene beads (Fluospheres™) that are pulled through the chip by a vacuum pressure-driven flow. The emitted photons from the beads were collected off the chip and after passing through a penta-bandpass optical filter (FF01- 440/521/607/694/809-25, Semrock) to remove excitation light, were detected by an avalanche photodiode (APD, Excelitas). A

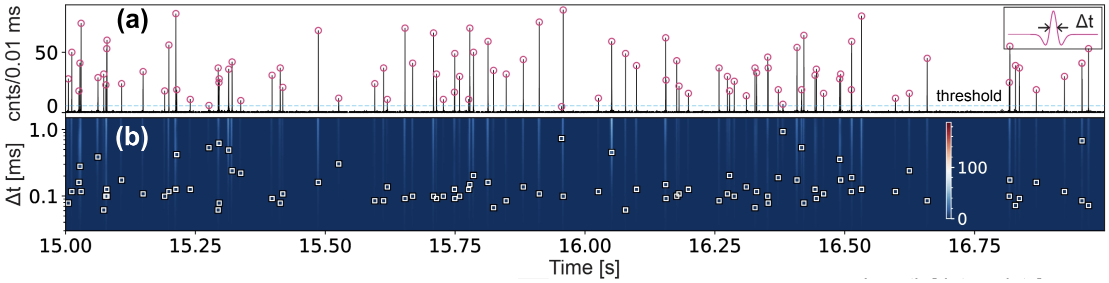


Figure 3.4: Parallel cluster wavelet analysis (PCWA) for single-peak analysis. (a) A cropped window of fluorescence signal taken from 200 nm fluorescent beads excited by single-mode (SM) waveguide (inset: Ricker wavelet used with PCWA algorithm). (b) CWT coefficients in time-Δt space (where a scaled and dilated version of the mother wavelet is convolved with raw data) with square markers indicating selected local maxima points found by the PCWA event detector algorithm.

single-photon-counting-module (TimeHarp 260 nano SPCM, Picoquant) records timestamps of the detected photons and stores them into a *.ptu file. A developed script adapted from [47] bins timestamps in 10 μs bins for further analysis and, therefore, large amounts of data points are generated in the memory. The two-second long trace displayed in Fig. 3.4a contains 200,000 points, and a 5 min acquisition time produces over 30 million points.

The signal height and width vary due to different particle positions in the fluidic channel [38] and fluctuations in flow speed, respectively. Previously, events were detected by setting a threshold of photon counts (background level) and signals above the background were counted as a particle [33]. While this works with bright targets and good chips, CWT analysis offers significant advantages in terms of accuracy, robustness, and information content. A Ricker mother wavelet (Mexican hat) is used to generate the 2D CWT coefficient map and is displayed as a color map in Fig. 3.4b. Only the positive part of the absolute value of CWT coefficients is kept to simplify the event localization step. Fluorescence signals in the real-time trace are now represented as bright streaks, and the Δt locations with the largest CWT coefficient correspond to the actual events and are highlighted with white boxes.

Fig. 3.5 illustrates a zoomed-in part of the trace to better understand how events are picked from the CWT map. First, local maxima (black dots with white borders in Fig. 3.5b) in the CWT map are identified at each scale (Δt value) with a conventional peak finding (with a user-defined threshold) process. This threshold is now in CWT space which is different from the threshold level shown in Fig. 3.4a. We first sort all CWT maxima by time and then define macro clusters (**MC**) local maxima that are separated by gaps along the t-axis by more than a predefined value (extent × 0.5 × $(N_j \Delta t_j + N_k \Delta t_k)$ for nearby j,k events). The 'extent' parameter is used to tune the threshold of **MC** formation and N is one for single-peak events. We then examine all clusters in parallel by calculating the distance values around the local maximum with the largest $C(t, \Delta t)$ value which is the first candidate for an event. This, too, is done in a parallel and vectorized fashion to speed up the event detection process. The overlap OL with other maxima within the cluster is

$$\begin{gathered}
OL(i,0) = sgn\left((r_i + r_0)^2 - d^2(i,0)\right) \\
d^2(i,o) = (t_i - t_0)^2 + (\Delta t_i - \Delta t_0)^2 \\
r_i = \frac{whN_i \Delta t_i \sqrt{C_i}}{\sqrt{w^2 {N_i}^2 \sin^2\theta_i + h^2 \cos^2\theta_i}}, r_0 = \frac{whN_0 \Delta t_0}{\sqrt{w^2 {N_i}^2 \sin^2\theta_i + h^2 \cos^2\theta_i}}, \\
C'_i = \frac{C_i - \min(C)}{\max(C) - \min(C)}
\end{gathered} \tag{3.3}$$

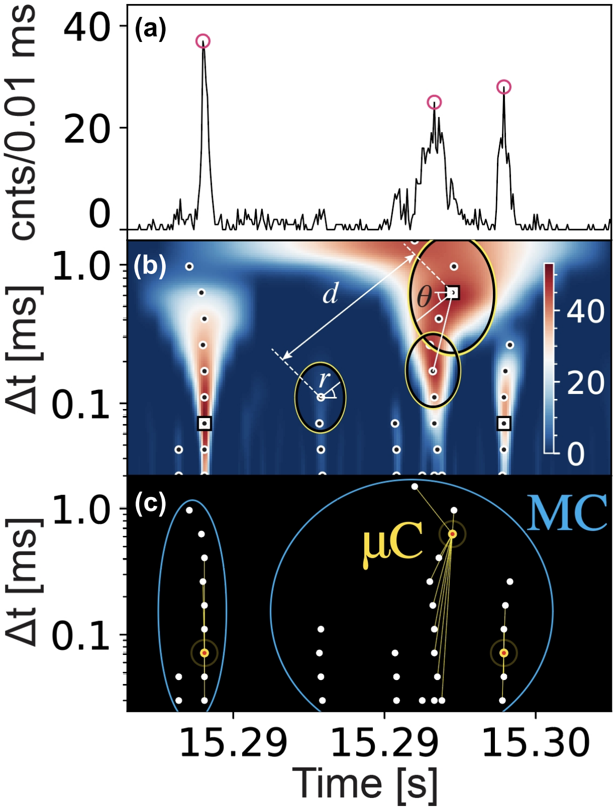


Figure 3.5: (a) Zoomed-in window of three events with circle markers showing the adjusted location of peaks. (b) CWT map of (a) including local maxima points (black dots). The clustering algorithm utilizes Euclidean distance and adjusted ellipses around each local maximum to search for links. The overlap of an ellipse with the centroid point defines a link. (c) Macro and micro-clusters ($\mathbf{M}C$ and μC): local maxima are first grouped into $\mathbf{M}C$ highlighted by blue circles by simplified 1D overlap calculation. The clustering algorithm finds μC for each $\mathbf{M}C$ in parallel. A μC is a star graph containing a minimum of links with the largest CWT coefficient maximum as the centroid (red-filled circles).

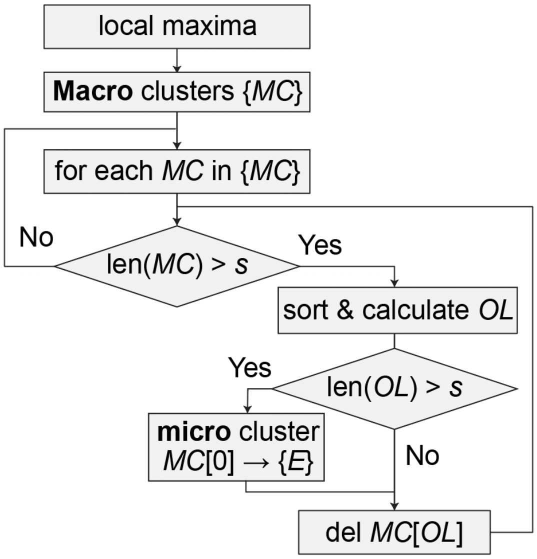


Figure 3.6: Flowchart of the clustering algorithm.

where r is the radius of the ellipses in Fig. 3.5b, d is the Euclidean distance, and w and h are adjustable spreading parameters that define the refinement sensitivity in time and scale, respectively. N represents the number of peaks in a multi-peak signal (here for a single-peak event $N = 1$). The addition of normalized CWT coefficient weights into $(r_i + r_0)^2$ helps detect weak events near strong ones. We then look for overlapping ellipses. If the number of points connected to the original largest maximum (centroid) is higher than a user-defined number (selectivity×len(scales)), it is taken as a micro cluster (μC) in which the actual event is immediately identified. The unconnected points form a new, smaller macro cluster where a new centroid is picked. The analysis repeats until no more clusters can be formed with a minimal, user-defined number of candidate points.

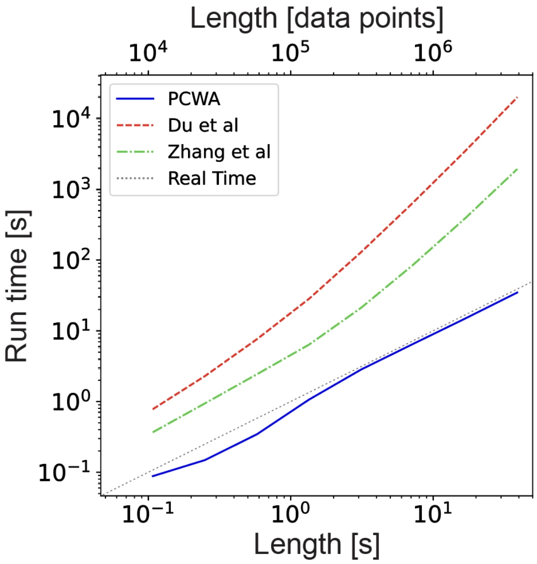


Figure 3.7: Run time comparison of clustering algorithm with established CWT peak finders, showing orders of magnitude faster speed and run times below the real-time limit (gray dashed line).

The outlined flowchart is rendered in Fig. 3.6. We call this algorithm parallel custom wavelet analysis (PCWA) due to the highly parallel fashion of processing data and customizable wavelet functions to match any shape of the signal [48].

Figure 3.7 compares the run time of the PCWA algorithm with the ridge-based CWT methods [45, 49] as a function of both the number of data points in the signal trace and the actual run time of the experiment. The comparison was done by running implementations of the algorithms in Python with 100 logarithmic scale values on a single desktop computer (Intel® Core™ i9-900 CPU with 32 GB of RAM). PCWA shows $O(N)$ complexity, where N is the number of data points. Consequently, it is orders of magnitude faster than other techniques and, most notably, PCWA always runs faster than the experiment itself (dotted line), e.g., 40 s of analysis for a 42 s trace, while other methods are impractical for real-time applications. This benchmark highlights the PCWA's capability for real-time analysis of time-dependent particle sensors.

3.3.1 Single-peak event detection

Single-peak event detection also called peak finding, is a very basic need in many applications. There are numerous tools developed to find peaks, and here we explore how PCWA can fulfill the peak-finding needs as well. In Fig. 3.8a, a zoomed-in window of a long APD trace is shown with five distinct peaks. These are 200 nm fluorescent beads. Here, we translate the scale to Δt value (see Fig. 3.8a inset) which is the full-width-half-max (FWHM) of the Ricker wavelet. This is comparable to the FWHM of the optical excitation spot inside the channel, thus directly convertible to the velocity of the detected particle. The join histogram visualized in Fig. 3.8c displays the correlation between the velocity and intensity of detected events. Since the intensity and velocity of flowing particles are tightly related to the position of the particle inside the channel, this joint plot helps to model the excitation region, as well as weeding out outliers as false events. Indeed, when compared to a conventional amplitude-based peak finder (with optimal threshold level set as seen as a dashed line in Fig. 3.8a), 5.7% more events were detected. In addition, the particles' velocity is also extracted from the scale values corresponding to the temporal widths of the fluorescence peaks.

We note again that the PCWA method is widely applicable to other sensor types. The example of single DNA detection with nanopore electrical sensor chips is discussed next. The particle dynamics are different in nanopore signals in comparison to

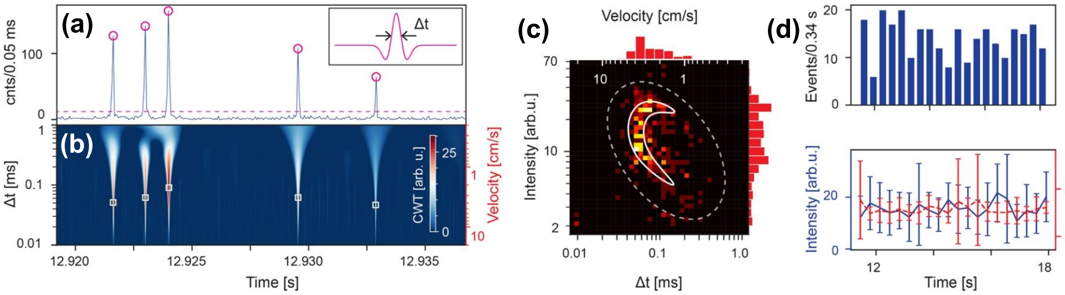


Figure 3.8: Single-peak event detection. (a) The zoomed-in window of fluorescence APD signal. (b) CWT coefficient scalogram with detected peaks shown as square boxes. (c) The joint plot of the intensity and Δt. (d) Time-varying information of events reflecting experiment dynamics. Error bars represent standard deviation.

pressure-driven fluorescence detection because the ionic current here is the driving force for particles [38]. Despite the source of the driving force, PCWA with a proper mother wavelet should enhance and detect matching events in a $(t,\, s)$ space. Nanopore sensors as ultrasensitive tools for the detection and analysis of individual nanoparticles are well-known for next-generation sequencing [50, 51, 52]. Individual particles moving through a nanoscopic membrane generate a characteristic change of an ionic current across the membrane. Similar to fluorescent signals, a time-dependent signal is produced where individual particles show up as a positive/negative peak also called translocation. The optofluidic chip with a modification is used as a nanopore sensor. Fig. 3.9a illustrates the experimental setup for the experiment where single SARS-CoV-2 RNAs were pulled through the nanopore by an applied voltage V_{NP}.

SARS-CoV-2 RNAs bounded to microbeads flow from reservoir 1 through the hollow-core waveguide and to the nanopore capture region by the applied electrokinetic voltage (V_{EK}). A trap-assisted capture rate enhancement (TACRE) technique [53] em-

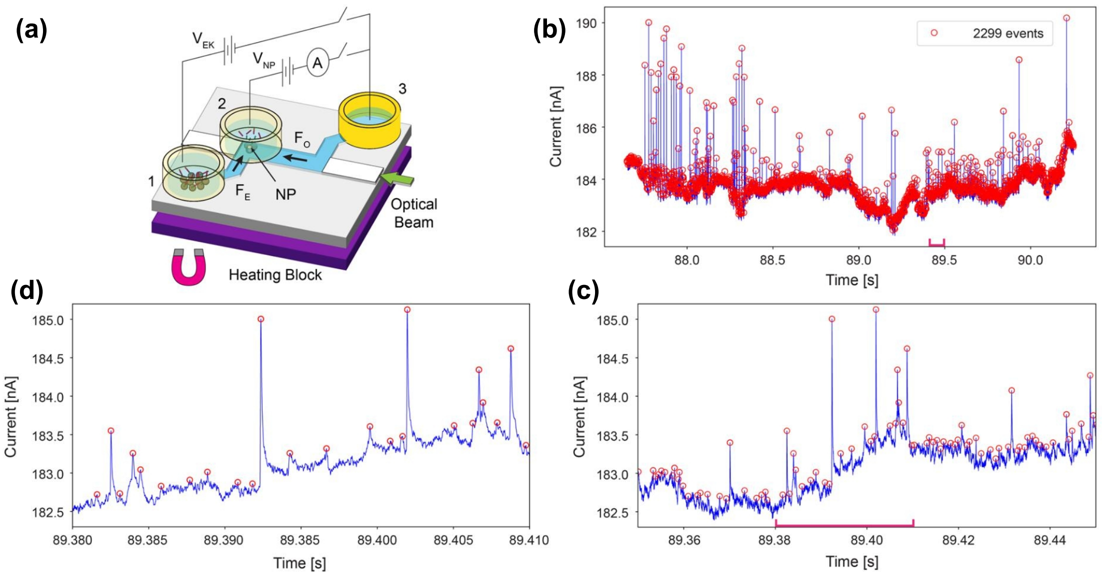


Figure 3.9: PCWA analysis of single-molecule nanopore sensor. (a) Schematic of the experimental setup to detect SARS-CoV-2 nucleic acids. (b) Full trace with $\sim$ 2,000 events detected in 2.6 s long trace. (c-d) Zoomed-in windows to show the location of the detected events using PCWA.

ploys the optical force from a light beam in the liquid-core waveguide to trap a group of microbeads underneath the nanopore. This increases the target concentration at the nanopore location and the rate of detection upon thermal release from the beads. Translocations of individual released nucleic acid molecules through the nanopore are detected from the current change between reservoirs (2) and (3). Signal shape, more specifically, blockade duration and depth depend on many factors such as the shape and location of DNA molecules. This variation of event signal is a time-frequency problem where a multi-scale CWT technique can improve detection accuracy significantly. Figs. 3.9b-d show how the PCWA algorithm detects the blockade events successfully without any additional pre-processing of the raw data such as baseline removal which is a major problem in nanopore signal processing. The high speed of the PCWA algorithm fits well for time-sensitive applications of nanopore sensing such as feedback-controlled, on-demand delivery of DNA molecules [54]. The accuracy of the PCWA method is benchmarked by the receiver operating characteristic (ROC) curve to incorporate true and false-positive events. In order to carry out a ROC curve analysis, a dataset with known positive events is required. Unfortunately, there is no ground truth event information for fluorescent- or nanopore-based signals. Therefore, we used a set of 75 mass spectrometry traces from a simulated protein spectra dataset [55]. Fig. 3.10 shows an example from the simulated spectra with a peak detection comparison of our PCWA with two other CWT-based methods. Gray vertical line markers indicate the true location of peaks and colored markers indicate the location of peaks detected by corresponding CWT-based peak finder methods. PCWA detected 84% of peaks correctly (62% for Du et al [45] and 86% for Zhang et al [49]) with 1% FDR (32% for Du et al and 7% for Zhang et al).

This dataset provides simulated protein spectra with noisy raw data alongside the true location of the peaks, which can be easily used to calculate the ROC curve. A ROC curve is a tool to visualize and compare the performance of a detector across a range of threshold parameters for the detector. The x and y axes of a ROC curve correspond to the true positive rate (TPR or recall) and false positive rate (FPR), respectively. A perfect detector is a single point in the top-left corner of the ROC space (TPR=1, FPR=0), detecting all true events (no false detection) independent from the set threshold. The area under the curve (AUC) for a ROC curve is a number between 0

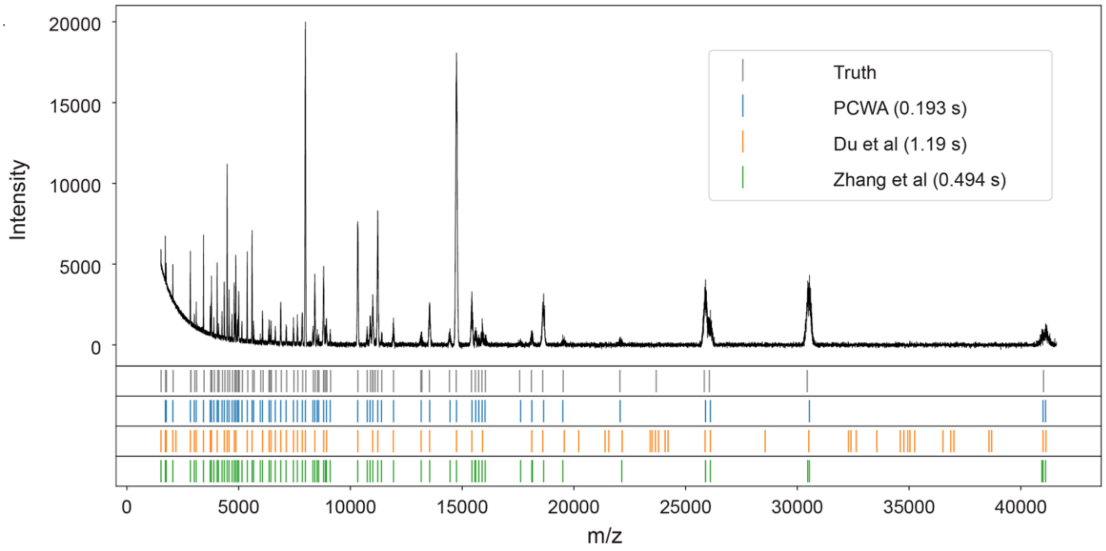


Figure 3.10: Mass spectroscopy peak detection. An example of peak detection done by three CWT methods on a simulated protein spectrum.

and 1 for normalized FPR and TPR and a higher AUC value means a better detector. The TPR and the FPR are calculated as

$$\begin{gathered} TPR = \frac{TP}{TP+FN}, \quad FPR = \frac{FP}{FP+TN}, \\ TP : \text{true positive}, \qquad FP : \text{false positive}, \\ TN : \text{true negative}, \qquad FN : \text{false negative} \end{gathered} \tag{3.4}$$

We define a false positive as a detected peak that is not located within ±1% of the mass-to-charge ratio (m/z) value of the true peak. Fig. 3.11 shows the average of ROC curves for 75 randomly selected simulated spectra and Table 3.1 summarizes the runtime comparison of the three methods. The typical size of an MS spectrum is $\sim$ 20,000 data points and 50 levels of logarithmic scale values were used for analysis. We observed that PCWA maintains excellent performance with significantly accelerated

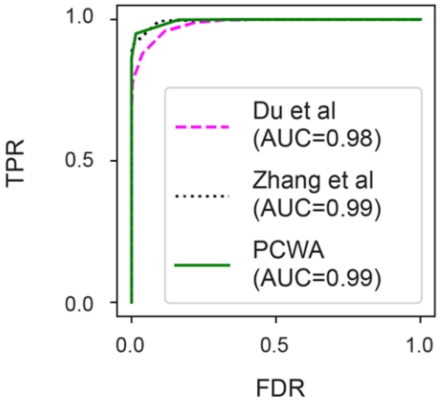


Figure 3.11: ROC plot generated by varying the threshold (minimum SNR for Du et al and Zhang et al methods). Each spectrum has ∼ 20,000 data points and 50 levels of logarithmic scale values are used to calculate CWT.

method	runtime [s]
Du et al [45]	0.96±0.12
Zhang et al [49]	0.44±0.04
PCWA [48]	0.23±0.15

Table 3.1: Runtime comparison of three CWT-based peak finders run on simulated mass spectra dataset.

analysis time.

3.3.2 Multi-peak event detection

Multi-peak signals offer significant advantages for practical sensing applications. With the addition of redundancy and encoding of temporal information to the signal, more reliable identification of events from a noisy background can be achieved. Multi-peak signals also enable multiplex detection if different targets produce different signal patterns which are particularly desirable for biomedical applications. This is usually done by the use of spatial excitation patterns using masks [56] or waveguides

[37, 57] that result in time modulation of the signal created by a particle crossing the excitation region. These can then be analyzed using Fourier transform analysis, shift-multiply algorithms (see below)[37], or matched filters. For example, signal-to-noise ratio (SNR) enhancement via multi-spot excitation implemented with Y-splitters and multi-mode interference (MMI) waveguides has shown up to $50,000\times$ SNR improvement [37, 57]. Signal analysis improvement with spatially encoded excitation has been explored in [58, 59, 60]. Here, we expand the application of the PCWA algorithm to multi-peak signals created by fluorescently tagged plasmid molecules corresponding to the Klebsiella pneumonia carbapenemase that cross the multi-spot excitation pattern inside the analyte channel. The chip design and experiment methodology are based on the multi-mode-interference (MMI) waveguide optofluidic biosensor discussed in [36, 4]. In this experiment, a HeNe laser (Melles Griot) working at 633 nm was coupled into a single-mode optical fiber-coupled (F-SA, Newport) and then into the chip. The single-mode waveguide guides light from the optical fiber and launches into the MMI waveguide (see Fig. 3.2a) while the rest of the setup is the same as the single-peak detection experiment. Figure 3.12a shows an example of a multi-peak signal from a single DNA molecule that consists of seven peaks representing the MMI waveguide excitation pattern defined at 633 nm. The Fourier transform of this signal shows a strong peak at $\sim$ 4.5 kHz which is related to the uniform spacing Δt of the seven subsequent peaks. In addition, the second harmonic at 9 kHz and strong content at very low frequencies are visible. Previously, multi-peak signals were detected and classified (by peak number) for multiplex detection using a shift-multiply algorithm [37, 57, 60] with good results.

The shift-multiply formula can be described as

$$S(t, \Delta t_C)_C = \left\{ \prod_{i=0}^{N_C-1} F(t - i \cdot \Delta t_C) \right\}^{\frac{1}{N_C-1}} \tag{3.5}$$

$F(t)$ is the fluorescence signal and N_C is the number of spots generated by the MMI waveguide at different wavelengths. This algorithm requires the detection of events and extracting approximate Δt_C value for each event. Then, the properly cropped window of the event is shifted by Δt_C for $N - 1$ times and multiplied by the original cropped signal to calculate $S(t)$ value for different Ns [61]. This is illustrated in Fig. 3.12b. The resulting product is large for a correct signal and very small for incorrect peak numbers or temporal spacing, resulting in significant SNR improvement compared to purely threshold-based counting [37, 57]. Despite the huge SNR improvement, this technique suffers from some practical challenges, such as, accurately determining the start and end of an event, or getting a larger value for bigger Ns in the presence of high background, which emphasizes the need for a robust multi-scale event detection algorithm. PCWA as our universal event detector, when combined with custom-designed wavelets exhibits a significant improvement in multi-peak event detection. We start with a mother wavelet that is commonly used in CWT data analysis for periodic patterns, the Morlet wavelet (Fig. 3.12c). It is a Gabor windowed (Gaussian function) sinusoidal (real Morlet) or exponential (complex Morlet) function defined as

$$\psi(t, \Delta t) = \begin{cases} \dfrac{1}{\sqrt{\sigma}} \exp(\dfrac{-t^2}{2\sigma^2}) \cos(\dfrac{2\pi}{\Delta t} t) & \text{real Morlet} \\ \dfrac{1}{\sqrt{2\sigma}} \exp(\dfrac{-t^2}{2\sigma^2}) \exp(j \dfrac{2\pi}{\Delta t} t) & \text{complex Morlet} \end{cases} \tag{3.6}$$

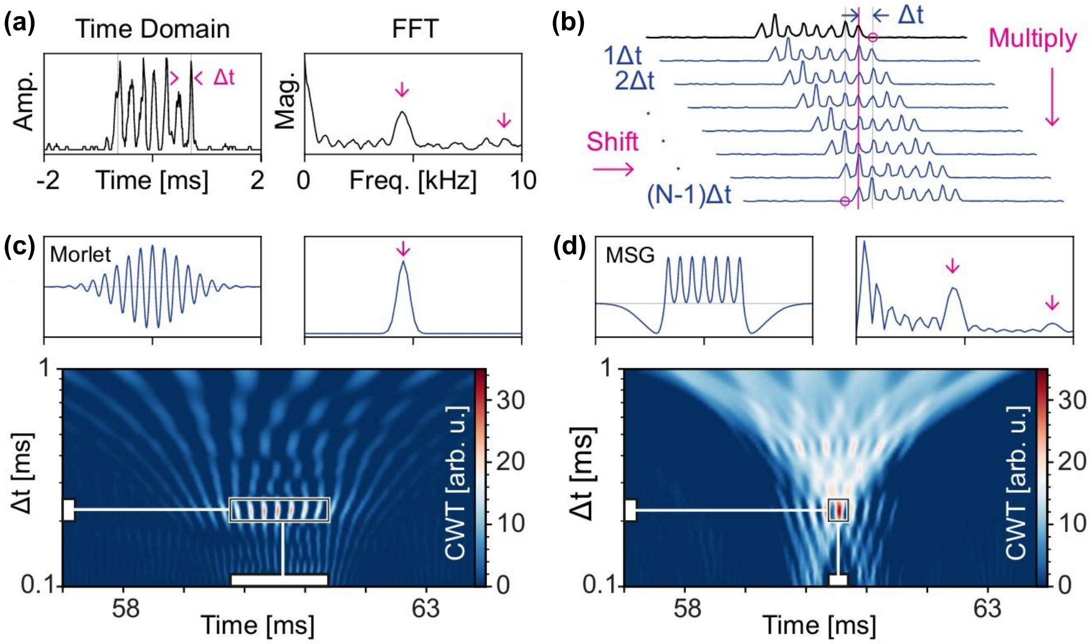


Figure 3.12: (a) Time and frequency domain representation of a fluorescence signal taken from single K. pneumoniae carbapenemase DNA molecules. (b) Shift-multiply algorithm previously used to find and classify multi-peak signals. (c) Time and frequency information of Morlet wavelet with single component aligned to the first harmonic of multi-peak signal. 2D map of positive CWT coefficients with white boxes indicating approximate certainty of time and scale localization. (d) Multi-spot Gaussian (MSG) wavelet designed to match the multi-peak signal as well as the full frequency spectrum; CWT coefficients map shows precise event localization.

where $\sigma \propto \Delta t$ defines the with of the Gabor window. The frequency spectrum from FFT calculation reveals a single and confined frequency component for the Morlet wavelet (Fig. 3.12c, top right) and if we use it with PCWA on the fluorescence signal, the main frequency is precisely localized as seen in the CWT map (Fig. 3.12c bottom). Multiple bright spots in the CWT map mean poor time localization. An approximate uncertainty box is drawn to reflect this problem for temporal information. On the other hand, we create a custom mother wavelet that resembles the signal pattern to be detected, here, the 7-peak MMI signal of Fig. 3.12a. This custom wavelet function is named Multi-Spot Gaussian (MSG) wavelet (Fig. 3.21d, top left) and is constructed by the sum of N Gaussians separated by Δt and encapsulated by two negative skewed peaks and mathematically described by

$$\begin{aligned}\psi_N(t,\Delta t) = &\sum_{n=0}^{N-1} \exp\left(\frac{-\left[t-\left(n-\frac{N-1}{2}\right)\Delta t\right]^2}{2\Delta t^2\sigma_+^2}\right)\\ &-\sum_{k=\pm 1}\frac{2a}{\Delta t\sigma_-}\phi\left(\frac{t+k\left(\sigma_- m_0-\frac{N}{2}\right)\Delta t}{\Delta t\sigma_-}\right)\Theta\left(k\alpha\frac{t+k\left(\sigma_- m_0-\frac{N}{2}\right)\Delta t}{\Delta t\sigma_-}\right)\end{aligned} \tag{3.7}$$

here, the σ_+ parameter for positive peaks is fitted to those measured from multi-peak signals normalized to Δt. a and σ_- are calculated according to positive peaks to achieve optimal temporal compactness and sensitivity. For higher sensitivity of the wavelet to N (specificity in multiplexed detection), the maxima (m_0) of the skewed Gaussian functions at both ends are placed at Δt from the first and last positive peaks. This ensures that the zero mean condition applies [46], and the wavelet is scaled for a square norm of one.

The parameters of the skewed Gaussian functions are explained in Eq. 3.8.

$$\begin{gathered}\phi(t)=\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{t^2}{2}\right),\Theta(t)=\int_{-\infty}^{t}\phi(u)du=\frac{1}{2}\left[1+erf\left(\frac{t}{\sqrt{2}}\right)\right]\\ m_0(\alpha)\approx\mu_z-\frac{\gamma_1\sigma_z}{2}-\frac{sgn(\alpha)}{2}\exp\left(-\frac{2\pi}{|\alpha|}\right)\\ \delta=\frac{\alpha}{\sqrt{1+\alpha^2}},\mu_z=\sqrt{\frac{2}{\pi}},\sigma_z=\sqrt{1-\mu_z^2},\gamma_1=\frac{4-\pi}{2}\frac{\left(\delta\sqrt{\frac{2}{\pi}}\right)^3}{\left(1-\frac{2\delta^2}{\pi}\right)^{3/2}}\end{gathered}\tag{3.8}$$

The Fourier transform of the MSG wavelet is shown in Fig. 3.12d (top, right) and shows excellent qualitative agreement with the FFT of the multi-peak signal from the KPC target. The improved time localization can be seen in the CWT map obtained with the MSG wavelet. It shows a single bright spot (black box) with a compact uncertainty box in time and Δt axes (white boxes), meaning that multi-peak events can be detected precisely. Figure 3.13a depicts again a subset of cropped windows from a 20-min long APD trace of the KPC sample. Using the MSG mother wavelet, we are able to detect multi-peak events by accurately locating them in time and frequency. The frequency (or scale) information is related to the velocity of the detected particles similar to single-peak event detection. The CWT map (Fig. 3.13b) shows clean identification of these signals and also illustrates the variation in Δt due to the different velocities of the molecules. Note that the gaps in the time axis of Fig. 3.13a, b were added solely to enable visualization of multiple plasmid detection events, which are very sparse at low concentrations. The ridge-based CWT methods would fail if we use our MSG wavelet applied to multi-peak signals. The issue can be seen in Fig. 3.13c where the local maxima (equivalent to Fig. 3.5c) around a single seven-peak event are shown. There are multiple maxima at each Δt level, creating multiple ridges and we will end

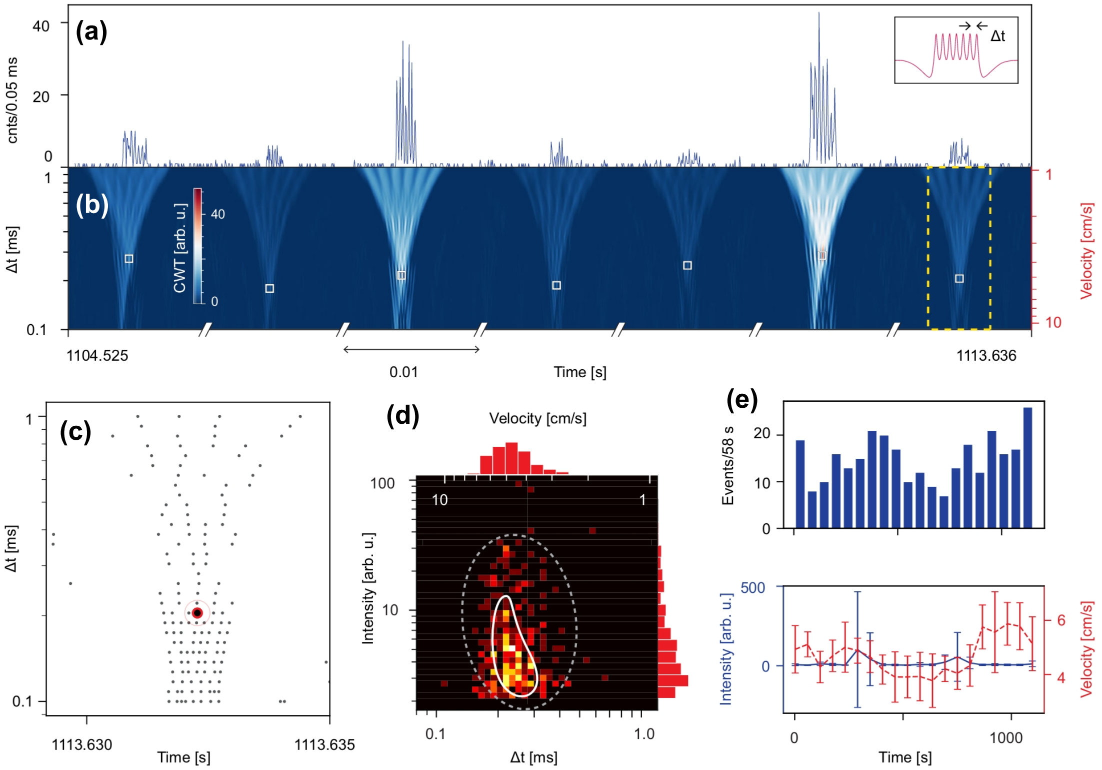


Figure 3.13: Multi-peak event detection. (a) Concatenated 10 ms cuts of detected single KPC molecules in a 9 s window. Inset is the multi-spot Gaussian (MSG) wavelet used to analyze the trace. (b) Corresponding time-Δt CWT scalogram with white squares showing detected multi-peak events across a range of Δt values. (c) Local maxima for a single example event render conventional ridge-line methods impractical. (d) Scatter plot for particle intensity and speed, showing a cluster in the predicted region (white line); dashed line: confidence region for event identification as DNA molecules. (e) Time-varying information of events during the measurement for dynamic determination of the flow characteristics. Intensity and velocity plots are the average value for events within a bin from the histogram plot. Error bars represent standard deviation.

up with multiple event detection for a single particle. In contrast, the cluster-based PCWA algorithm addresses this problem. A single micro cluster will be formed for these local maxima and can be quickly identified as a single event. In Fig. 3.13d, the intensity and Δt (velocity) of all 300 detected molecules are visualized in a 2D histogram to give joint information in a single plot. The distribution is determined by the waveguide mode profiles and the (parabolic) velocity profile inside the microchannel. The predicted distribution of the majority of events for the chip under consideration is shown as a white line and matches the data well (see Appendix A for more detail). This helps to further polish the detection list by rejecting those sitting outside the kernel. Figure 3.13e shows the progress and dynamics of the particles inside the fluidic channel during the experiment. Good agreement of velocity fluctuations and detection rate is observed. Real-time analysis was explored in different experiments by utilizing PCWA to implement the event detection algorithm on a sliding window of the buffered signal during data acquisition. Such real-time statistics can help monitor proper experimental operation, e.g., maintaining constant vacuum pressure. The comparison of three main methods for multi-peak event detection, Shift-Multiply, PCWA (Morlet), and PCWA (MSG), is done by evaluating the detection rates and their accuracy. Individual events are inspected manually to tell true from false detections. The accuracy is defined by the algorithms' ability to classify the correct number of peaks (here: seven). This is useful in multiplexed detection with MMI waveguides [36, 4]. Three numbers (6, 7, 8) are selected as the possible output of classifiers. For PCWA, the output N is determined by applying all three MSG- wavelets (MSG-6, MSG-7, MSG-8) simultaneously and letting

	Shift-multiply	PCWA (Morlet)	PCWA (MSG)
Detection rate [$\times 10^4$ events/mL]	1.03	1.84	4.60
Accuracy [%]	66.37	37.04	94.03

Table 3.2: Detection rate and accuracy comparison of Multi-peak event detection methods.

the clustering algorithm pick the strongest candidate. In the beginning, a series of threshold values were scanned for a random small part of the fluorescence signal to find the optimal compromise between accuracy and detection rate. The methods comparison is presented in Table 3.2 and the significant improvement achieved by the MSG mother wavelet outperforms the other two methods.

With over 4× more identified events and 6× fewer errors than the shift-multiply algorithm, PCWA is advised to be used instead of shift-multiply in optofluidic fluorescence single-particle detection. The lack of ground truth in this experiment pushed us to generate a similar time series with a known list of events. This simulation trace, just like the simulated mass spectra, increases confidence with the benchmarking analysis. Figure 3.14 shows examples of two data traces, Fig. 3.14a for the KPC fluorescence signal, and Fig. 3.14b simulated trace to mimic similar signal characteristics. Simulated events are first carefully created by placing 7 Gaussian functions with spacing and width similar to the real KPC signals. Next, these template multi-peak signals, are randomly stretched and amplified to fit with the joint distributions data extracted from the KPC data trace. Eventually, a constant background signal is added and the entire signal is passed to a Poisson function. Poisson function mimics the behavior of the SPCM detector [62, 63]. The insets show a random multi-peak event picked for each

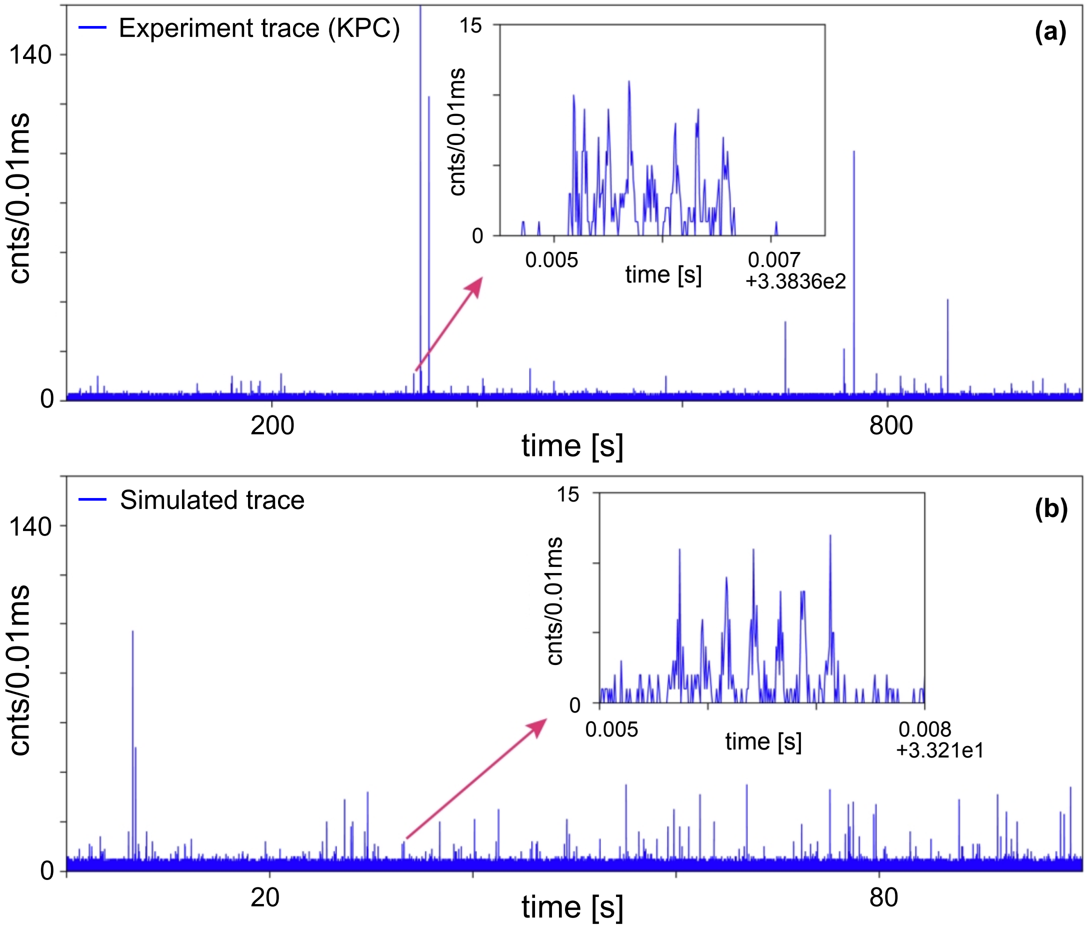


Figure 3.14: Experiment and simulated traces. (a) The fluorescence signal recorded from the KPC detection experiment (Fig. 4, main manuscript). Inset shows a randomly selected event. (b) The simulated trace with similar joint amplitude-velocity distribution with added noise. Inset shows a zoomed-in window of one of the 239 events.

signal. The benchmark results presented in Table 3.3 for the simulated data show a close match with the real-world data. This demonstrates the ability of the PCWA method to instantly extract additional valuable information from the sensor data. Because the optical excitation patterns are generated by lithographically defined waveguides according to the MMI principle, they have well-defined physical spacing.

In Figs. 3.15a-c, a 100 s window of the simulation trace is shown with detected events shown as upside-down triangles. Ground truth events are unfilled circles and a-c are the results of event detection for PCWA (MSG), PCWA (Morlet), and shift-multiply respectively. We can see a bigger population of red markers (true events) for PCWA (MSG) with 204 out of 236 detected events identified correctly. The other methods detect only a small count of events from a total of 239 randomly distributed events in the simulation trace. Figs. 3.15d-f shows a zoomed-in window as an example of the weak signal at the background level and successfully detected and identified by PCWA (MSG). Shift-multiply fails to even detect the event due to a higher background threshold set to avoid random background signals getting picked as events. PCWA (Morlet), even though it detects weak signals, fails to correctly identify the N because of vanishing sides and less sensitivity to N.

3.3.3 Multi-speed multiplexing in PDMS devices

In another MMI-based multiplexing approach, we utilized a multi-channel concept to flow two separate solutions at different velocities [10]. A planar waveguide device was fabricated using polydimethylsiloxane (PDMS) elastomer using a standard

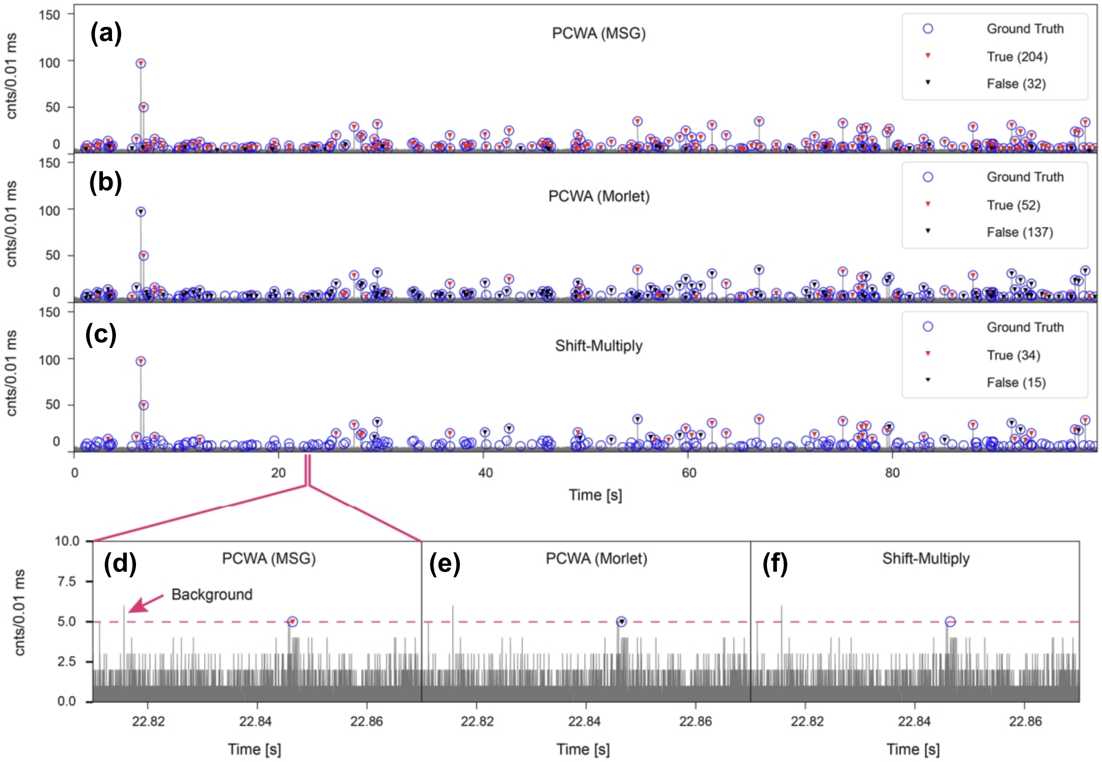


Figure 3.15: Performance comparison of three methods on the simulated trace. (a) Events detected using PCWA with MSG wavelet. Most of the events (>98%), even those buried in the background were accurately identified (86%). Only one false event from noise was picked as an event. (b) Detection was done by PCWA with a Morlet wavelet. Compared to MSG wavelets, significantly less accurate localization of events with more FDR is observed. The accuracy of identifying the type of target is the worst when using a Morlet wavelet. (c) Detected events using the Shift-Multiply technique. Comparable background level to weak events prevents accurately locating and identifying events. (d-f) examples of a weak signal analyzed by three methods; only PCWA (MSG) is able to precisely locate and identify it.

	Shift-Multiply	PCWA(Morlet)	PCWA(MSG)
total detected	49	189	236
TPR	0.205	0.782	0.983
FDR	0.000	0.011	0.004
accuracy(%)	69.4	27.5	86.4

Table 3.3: Performance comparison of Shift-Multiply and PCWA methods with simulated fluorescence signal. The metrics are measured based on the ground truth event list.

soft-lithography recipe [64, 65]. By controlling base:curing agent ratio during the fabrication, we can alter the optical properties of PDMS, specifically the refractive index [65]. The other technique to control refractive index is by controlling curing temperature and period [11], however, we used the former method due to the simplicity of the process. Fig. 3.16a shows the cartoon of a double-channel device featuring a liquid-core MMI waveguide (75μm wide and 1.86 mm long) for excitation and a solid-core collection waveguide to collect emitted photons from fluorescent particles. The input light from an optical fiber gets coupled into a single-mode waveguide which then launches into the MMI waveguide. The MMI waveguide is filled with ethylene glycol (≈ 1.431 [66]) which has a refractive index slightly higher than 5:1 PDMS (≈ 1.415 [65]) to provide optical guiding. All waveguide dimensions are derived from numerical simulations (using FIMMWAVE/FIMMPROP) such that integer numbers of bright spots form in Ch_1 and Ch_2 at three wavelengths of interest (λ_{in} = 633 nm, 556 nm, and 488 nm). Fig. 3.16b shows an example of the simulation result for λ_{in}=633 nm offering 7 bright spots. 8 and 9 spots are expected at green and blue lasers. The optical microscopy image shown in Fig. 3.16c is from the fabricated device at the Jack Baskin School of Engineering (JBSOE) at UCSC. Figs. 3.16d-f are fluorescence images taken from the analyte channel when Ch_1 and Ch_2 are both filled by a mixture of DI water and quantum dots. The colormaps are set according to the input laser light. We can see predicted N bright bands at Ch_1 extend to Ch_2, meaning that fluorescent beads flowing in Ch_1 and Ch_2 will create the same number of peaks in the APD signal. We pulled a solution with a mixture of polystyrene beads through the analyte channels at different rates, -35 inHg and -10 inHg for Ch_1 and Ch_2 respectively. Figure 3.16 is adapted from [10] with permissions.

The fluorescence signals are collected by an SPCM and recorded on a disk for post-processing. Photon timestamps are binned at 0.01 ms bins and visualized in Fig. 3.16g with four sections corresponding to four excitation scenarios. The colored bars on the top of each section indicate the laser used to excite flowing fluorescent beads. We choose the last section where all three lasers were turned on, therefore, all possible events exist in this section making the data analysis challenging. The PCWA program with three mother wavelets (MSG-7, MSG-8, and MSG-9 manually tuned to match the shape of events of each color) was applied to the selected window to detect and identify present events. Fig. 3.17a shows a zoomed-in part of the APD signal where we can see a few examples of multi-peak events from both channels. Events from Ch_1 are narrow in the time domain, because of their high velocity (fast events), and events from Ch_2 are much wider due to slow flowing speed (slow events). The multi-scale event detection with PCWA was able to locate most of the events in time-scale space (see Fig. 3.17b) with precise time and scale localization. Events from Ch_1 and Ch_2 are easily distinguished by looking at separate ranges of scale, fast events populating at the lower region, and slow events populating at the upper part of the scalogram. The matched filter behavior paired with the multi-scale localization of CWT offered in PCWA as a single fast data analysis framework is well suited for this velocity multiplexing concept.

Since Δt and the intensity of detected events are extracted inherently during event detection, we can do some more statistical analysis such as a joint histogram

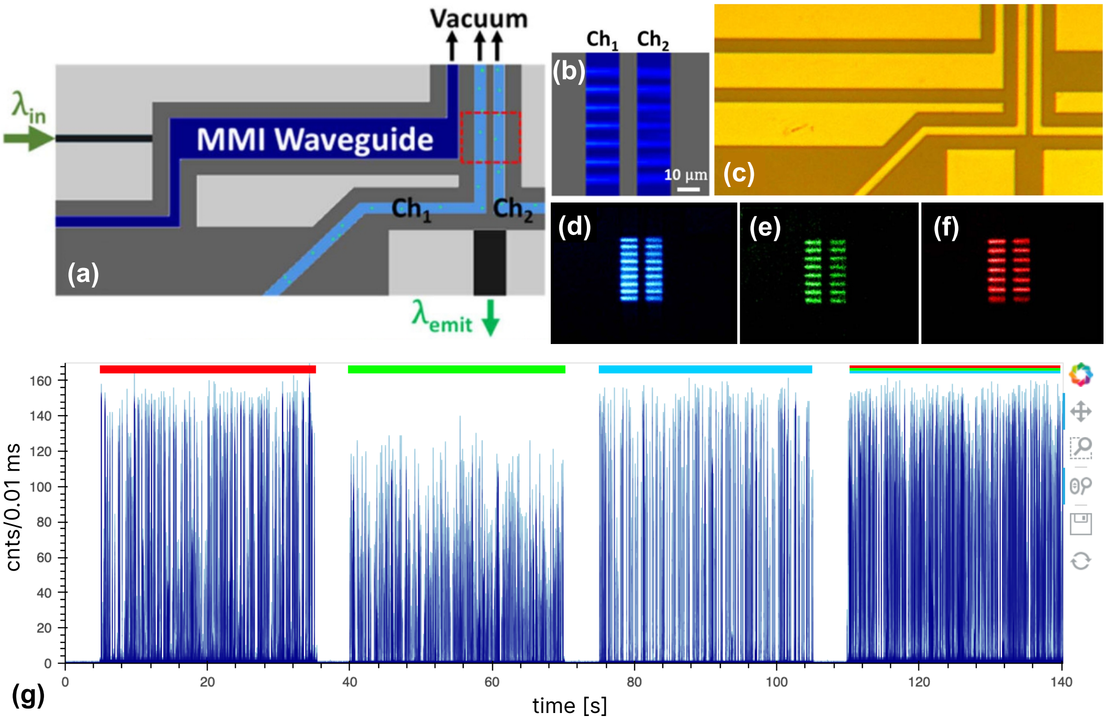


Figure 3.16: (a) Cartoon of the device layout. Input light from optical fiber (λ_{in}) gets coupled into the chip through the single-mode solid-core waveguide. λ_{in} is guided to an ethylene glycol-filled MMI waveguide, which creates a wavelength-dependent spot pattern at analyte channels (Ch_1 and Ch_2). Fluorescent particles are pulled through the excitation volume via variable negative pressures (vacuum) allowing control of the particle velocities. λ_{emit} is collected via a perpendicular multimode solid-core waveguide (black). (b) Numerical simulation predicts N=7 pattern at Ch_1 and Ch_2 for λ_{in} = 633 nm. The pattern seen at Ch_1 is preserved for Ch_2. (c) Optical microscope image of an unfilled device with a 50 μm wide MMI waveguide. (d-f) shows top-down fluorescent images to characterize the MMI waveguide performance at various wavelengths, Ch_1 and Ch_2 are filled with quantum dots (emission wavelength 665 nm with λ_{in} = 488, 556, and 633 nm). (g) APD trace from 1 μm diameter fluorescence polystyrene beads flowing through Ch_1 and Ch_2 simultaneously by -35 inHg and -10 inHg vacuum pressure respectively. Four sections marked by colored bars on top indicate the active laser source(s) $\lambda_{in(s)}$.

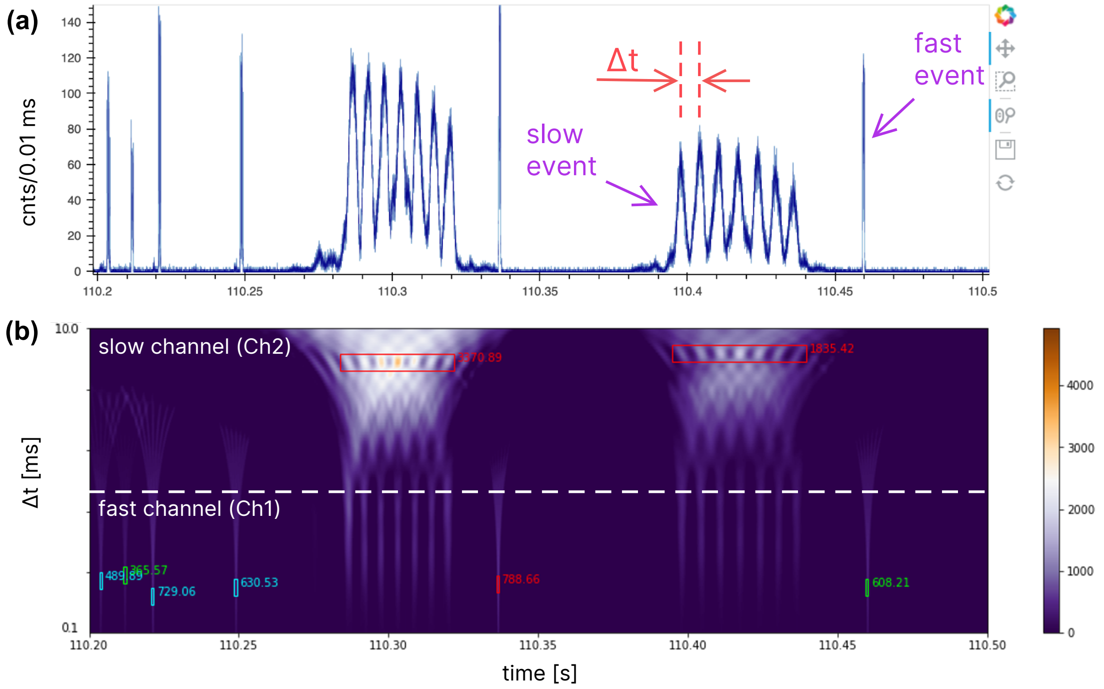


Figure 3.17: (a) A cropped window of APD signal shown in Fig. 3.16g when all three lasers are on. There are multiple fast and slow events corresponding to fluorescent beads from Ch_1 and Ch_2 respectively. (b) CWT scalogram of the cropped window for MSG-7 wavelet (only one scalogram can be visualized at a time) is shown for a large Δt range (0.1-10 ms). Different events from different colors of fluorescent beads are accurately located and identified using the PCWA algorithm when we apply three corresponding wavelets (MSG-7, MSG-8, MSG-9) shown with red, green, and cyan boxes. The CWT coefficient of individual events is printed next to each box and the width of the box is the approximate time length of the event ($N \times \Delta t$). The white dashed line separates the scale band for slow and fast events making multiplexed detection of events from separate channels possible.

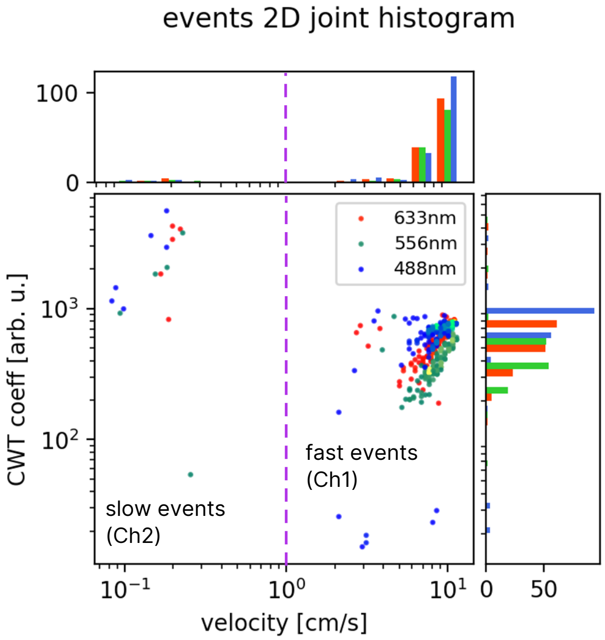


Figure 3.18: 2D joint histogram visualizes the scatter plot of detected events in velocity and intensity (CWT coeff.) space. The velocity histogram depicts two easily separable populations of events for Ch_1 and Ch_2. The number of events for Ch_2 is significantly smaller than for Ch_1 because of the slower flow rate.

(Fig. 3.18) to explore more information about the sample. The 2D joint histogram plot for detected events reveals that velocity distributions for fluorescent beads flowing in Ch_1 and Ch_2 are almost a couple of orders of magnitude apart and, thus, easily separable. The average velocities are 8.67±1.77 cm/s and 0.167±0.05 cm/s for Ch_1 and Ch_2 respectively (± standard deviation). The total number of slow events is 16 (R:5, G:5, B:6) while we see 427 (R:140, G:127, B:160) for the fast ones which makes sense with the velocity differences between the two channels.

3.4 Toolboxes to use PCWA (script and GUI)

In order to ease the analysis for end-user, especially during an experiment and when there is a need to do a quick analysis, we have developed a couple of tools. The base source code is written in an objective-oriented programming (OOP) scheme, to support any future development and embedding of PCWA in our projects. PCWA repository at GitHub (https://github.com/vganjali/PCWA) includes the source code and instructions to install and use in a Python script. Below are the required packages with minimum versions tested to run PCWA:

- Python $\geq$ 3.8.5
- numpy $\geq$ 1.19.2
- scipy $\geq$ 1.6.2
- matplotlib $\geq$ 3.3.4
- h5py $\geq$ 2.10.0
- pandas $\geq$ 1.2.1

PCWA is published and maintained at the Python Package Index (PyPI) repository and can be installed on a machine by using Python's PIP package manager and calling `pip3 install --upgrade pcwa` in the terminal. The example below shows how to use the PCWA event detector to detect peaks in mass spectroscopy data.

```python
import numpy as np
import pandas as pd
import pcwa as pcwa
import matplotlib.pyplot as plt

# read the raw mass spectroscopy data and truth values
df_raw = pd.read_csv('n100sig66_dataset_1_25/Dataset_14/RawSpectra/noisy22.txt',sep=' ')
df_true = pd.read_csv('n100sig66_dataset_1_25/Dataset_14/truePeaks/truth22.txt',sep=' ')

# create pcwa_analyzer object and set the desired parameters
pcwa_analyzer = pcwa.PCWA()
pcwa_analyzer.trace = df_raw['Intensity']
pcwa_analyzer.dt = 1
pcwa_analyzer.scales = [10,100,100]
pcwa_analyzer.wavelet = ['ricker']
pcwa_analyzer.keep_cwt = False
pcwa_analyzer.w, pcwa_analyzer.h = 0.2, 1
pcwa_analyzer.show_wavelets = False
pcwa_analyzer.use_scratchfile = False

# detect events (peaks)
events = pcwa_analyzer.detect_events(threshold=200)

# fine tune the location of detected peaks
loc = [int(e-events['scale'][n]+np.argmax(df_raw['Intensity'][int(e-events['scale'][n]):int(e+events['scale'][n])])) for n,e in enumerate(events['loc'])]

fig, ax = plt.subplots(3,1,figsize=(16,4),dpi=96,sharex=True,gridspec_kw={'height_ratios': [12,1,1]})
plt.subplots_adjust(hspace=0,wspace=0)
l0, = ax[1].plot(df_true['Mass'],df_true['Particles']*0, '|',markersize=10,color='gray',label='Truth')
ax[0].plot(df_raw['Mass'],df_raw['Intensity'],color='blue')
l1, = ax[2].plot(df_raw['Mass'].iloc[loc],[0]*len(loc),'|',markersize=10,color='red',label='PCWA')
ax[1].set_yticks([])
ax[1].set_ylim(0,0)
ax[2].set_yticks([])
ax[2].set_ylim(0,0)
ax[0].set_ylabel('Intensity')
ax[-1].set_xlabel('m/z')
plt.legend(handles=[l0,l1], bbox_to_anchor=(1.0, 4), loc='upper left')
plt.show()
```

Calling this snippet plots the graph shown in Figure 3.19 with raw data, ground truth, and detected peaks all in one view. Despite the unlimited flexibility attainable by using scripts and generating any kind of plots, sometimes, especially during the experiment, a robust and easy-to-use tool is preferred. A graphical user interface (GUI) has also been developed to be used during the experiment, this allows for quick analysis of recorded data traces. The application is named single-molecule-detection (SMD) Analysis and the main parts of the program are shown in Fig. 3.20. The time trace view plots the binned data trace (bin size can be changed at the bottom of the control panel), as well as the detected markers with colored markers. At the bottom, there is a window to visualize the CWT time-scale 2D plot, also called a scalogram, in which enhanced events show up as bright regions depending on the wavelet used for analysis. Events are localized in the time-scale space by colored markers for better insight into detected events. The control panel on the left includes the necessary tools and configurations for PCWA analysis. At this time, the most updated features and recent changes are only available in the non-GUI version explained earlier, but the GUI helps to explore parameter effects and fine-tune the analysis settings. The control panel shown in Fig. 3.21 gathers four main tabs: File Explorer, Target, Wavelet, and Analyzer to configure and run the analysis. File Explorer provides a navigator to explore and select data traces (see Fig. 3.21a). In the target tab, the user can add/delete target types with related properties such as name and color for multiplexed target detection and identification purpose (see Fig. 3.21b). In Fig. 3.21c, wavelet settings can be adjusted with a preview of the wavelet assigned for a target. The wavelet shape can be customized via input

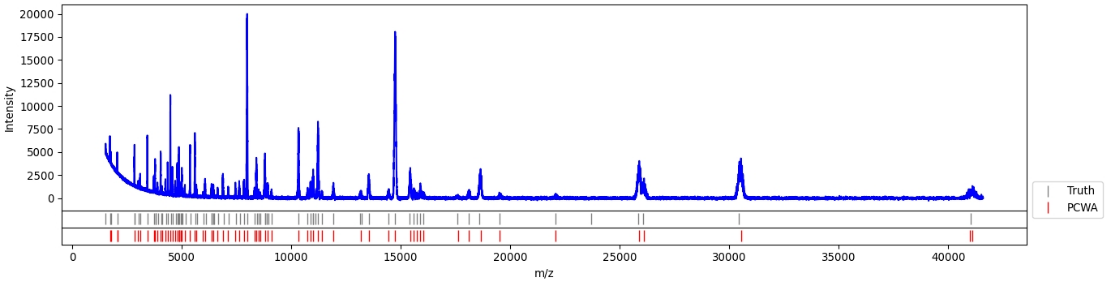
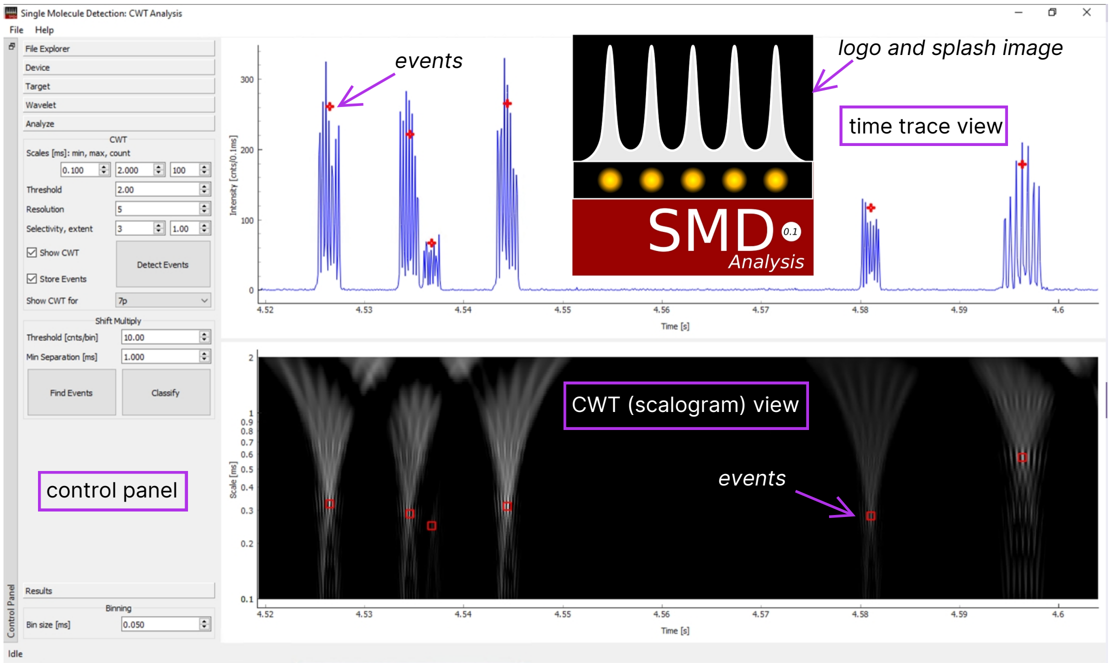


Figure 3.19: Mass spectroscopy peak detection using PCWA with example Python script.

widgets accessible in this tab. Finally, the PCWA event detection parameters such as scale range, threshold, resolution, and selectivity can be set in the Analyzer tab (Fig. 3.21d) and Detect Event button will start the analysis. If the “Show CWT checkbox” is checked, the CWT 2D plot will be displayed for the selected target (see “Show CWT for” dropdown box). Once the analysis is completed, the results are generated at the Results tab in addition to the markers of detected events in data trace and CWT viewboxes. In Fig. 3.22 different statistical graphs are depicted to help with the insight into the experiment. Detected events can be saved into a *.csv file for further studies in different programs like Matlab, MS Excel, etc.

These tools and GUIs are developed with open-source design and users can change the source to fit any specific problem and experiment.


Single Molecule Detection: CWT Analysis

File Help

File Explorer

Device

Target

Wavelet

Analyze

CWT

Scales [ms]: min, max, count 0.100 2.000 100

Threshold 2.00

Resolution 5

Selectivity, extent 3 1.00

☑ Show CWT

☑ Store Events

Show CWT for 7p

Shift Multiply

Threshold [cnts/bin] 10.00

Min Separation [ms] 1.000

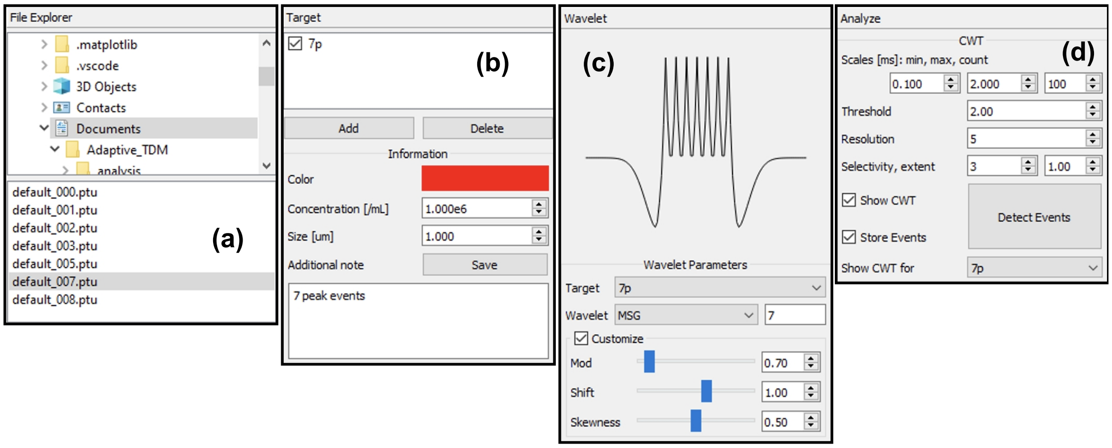
Results

Binning

Bin size [ms] 0.050

Idle

Figure 3.20: GUI developed using PCWA Analysis for single-model-detection experiments (SMD Analysis). Time trace and CWT views provide access to the data trace with the zoom feature. The binning for *.ptu files can be set via the "Binning" input box at the bottom of the "control panel". Other settings and analyses are accessible in the control panel. The file menu can help with saving and loading settings and configurations.


File Explorer

- .matplotlib
- .vscode
- 3D Objects
- Contacts
- Documents
  - Adaptive_TDM
    - analysis

default_000.ptu
default_001.ptu
default_002.ptu
default_003.ptu
default_005.ptu
default_007.ptu
default_008.ptu

Target

☑ 7p

Information

Color 🟥

Concentration [/mL] 1.000e6

Size [um] 1.000

Additional note

7 peak events

Wavelet

Wavelet Parameters

Target 7p

Wavelet MSG 7

☑ Customize

Mod 0.70

Shift 1.00

Skewness 0.50

Analyze

CWT

Scales [ms]: min, max, count 0.100 2.000 100

Threshold 2.00

Resolution 5

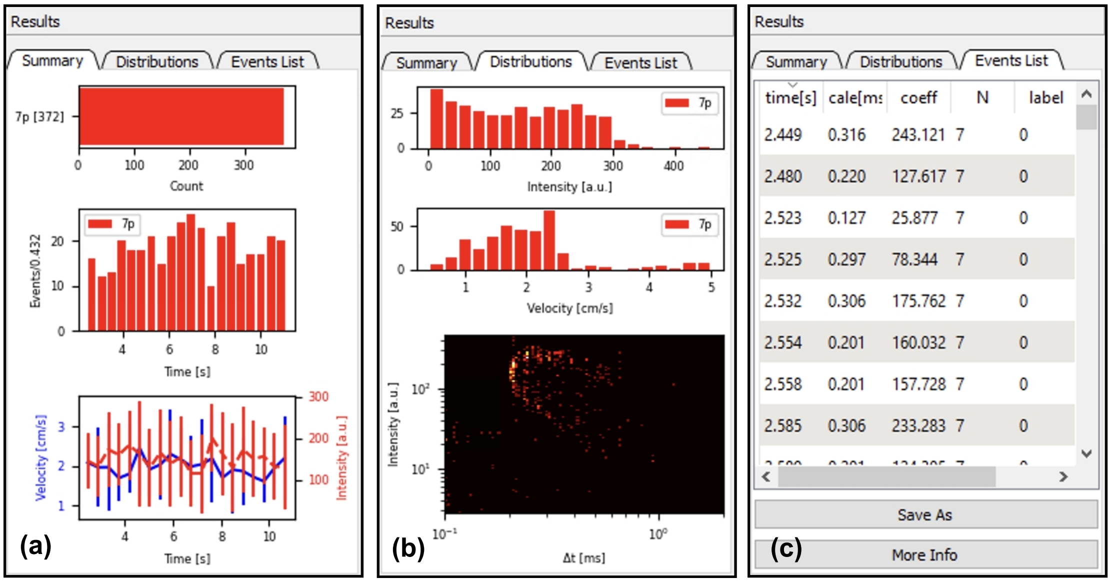
Selectivity, extent 3 1.00

☑ Show CWT

☑ Store Events

Show CWT for 7p

Figure 3.21: Important parts of SMD Analysis. (a) File Explorer: navigate and browse files. (b) Target: define target properties such as name, color, and some additional notes. (c) Wavelet: choose and customize the wavelet for every target. (d) Analyzer: here you enter the PCWA event detector settings.



time[s]	cale[ms	coeff	N	label
2.449	0.316	243.121	7	0
2.480	0.220	127.617	7	0
2.523	0.127	25.877	7	0
2.525	0.297	78.344	7	0
2.532	0.306	175.762	7	0
2.554	0.201	160.032	7	0
2.558	0.201	157.728	7	0
2.585	0.306	233.283	7	0

Figure 3.22: Important parts of SMD Analysis. (a-c) Results: various statistical information for all detected events can be discovered here. For further analysis, you can save events into a *.csv file via the “Save As” button in the “Events List” tab (see (c)).

Chapter 4

Machine learning-based pathogen identification

In this chapter, we will explore the use of machine-learning techniques to improve optofluidic-based pathogen detection and identification. This includes dataset preparation from real-world experiments, support vector machine and neural network classification models, training and testing models, and implementing a trained neural network on an edge device.

4.1 Introduction

The concepts and theories of machine learning and artificial intelligence (AI) have been around for many decades, if not a century, with the model of nonlinear neurons forming a network back in the mid-1990s [67]. Better learning theories were developed in the late 1990s, making it a functional tool in pattern recognition [68, 69].

Although the ML concept is not new, the field has exploded in the last two decades due to large-scale high-performance computation, which can handle ever-larger amounts of data. AI has become a part of our daily life and is used in numerous areas, including agriculture, environmental studies [70], autonomous vehicles, economics [71], marketing [72], drug development [73], entertainment [74], and medical diagnosis [75]. In biomedical diagnosis, artificial intelligence-assisted methods at different levels, from large-scale population data down to sensory data analysis, have been developed, underscoring the advantages of novel machine-learning over classical techniques [76, 77, 78]. Moreover, AI-assisted approaches have shown great potential in noninvasive automated diagnosis [79] and segmentation of multimodal brain images [80].

All these techniques rely on the fundamental idea of fitting a very complex mathematical model into the problem of interest without doing the complicated study and analysis of the model. Classically, problems were modeled by studying the underlying physics and finding the mathematical expression relating the inputs of the system to the outputs. With the bloom of machine learning, any given system can now easily be fitted into a model which is capable to do the same input-output mapping. This has become possible by access to gigantic datasets (or collecting and generating a new dataset) to train a model by use of powerful computational infrastructures. If we want to simplify the machine-learning problems, we can list them as

Supervised learning: training is done with labeled datasets.

- Classification

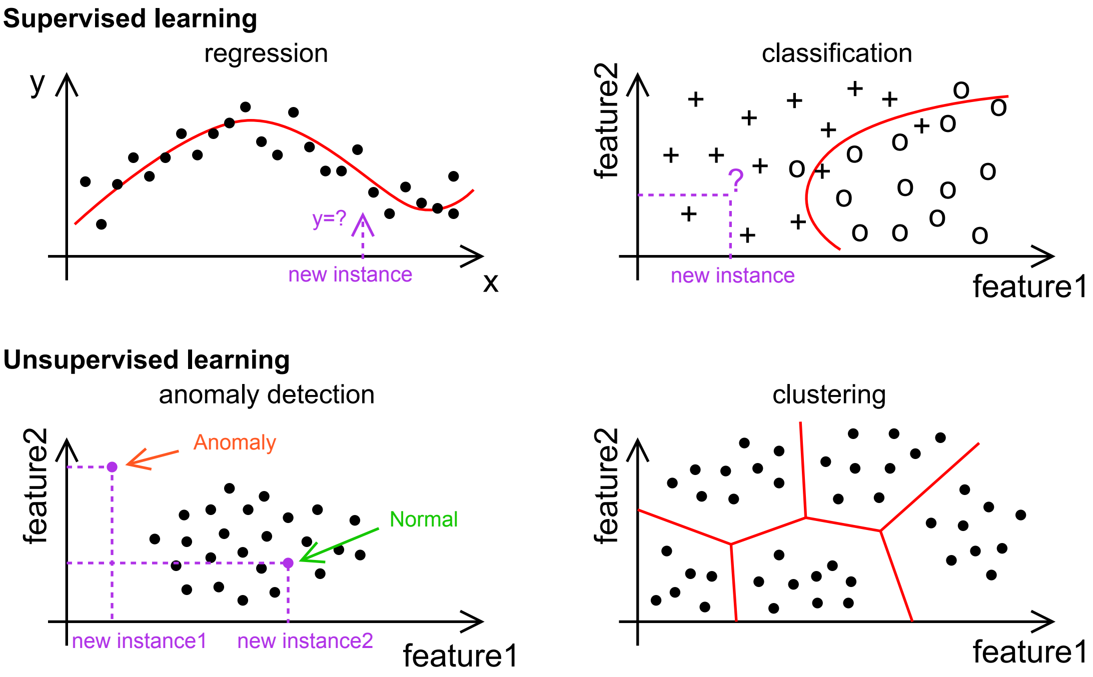


Figure 4.1: Examples of supervised and unsupervised learning.

- Regression

Unsupervised learning: training is done with unlabeled datasets.

- Clustering
- Dimensionality reduction
- Anomaly detection

Fig. 4.1 shows a couple of examples from supervised and unsupervised learning (adapted from [81]). In this chapter, we focus on supervised learning for classification applications to classify detected events from time series.

4.2 Dataset preparation

Data, as a fundamental element of machine learning, should be properly gathered and labeled. Here, we create the dataset from cropped events detected from fluorescence signals.

4.2.1 Experiment

Fluorescence signals are collected from tagged biomolecules and converted into electrical signals using an avalanche photodiode (APD). We use the same single-molecule-detection (SMD) setup discussed earlier in 3. Fig. 4.2b shows the experimental SMD setup used for this chapter with just a change in the lasers in comparison to the setup used in Chapter 3. Fig. 4.2a, illustrates the sandwich assay used as the target for detection. Our framework can be applied to any time-series signal that features different time signatures for different targets. We introduce it using a fluorescence detection assay for Klebsiella pneumoniae bacteria nucleic acid biomarkers.

The assay for event detection involves a sandwich design, where the target DNA strands are bound to a pulldown sequence on a streptavidin-coated microbead and then labeled with fluorescent probes in two colors, enabling detection under both red and green excitation wavelengths. This allows for three possible fluorescence signals (red only, green only, and red + green) depending on the excitation wavelengths used. To prepare the sample, 10 μL of 3 μM synthetic nucleic acid strands corresponding to the antibiotic-resistant bacterial target are mixed with 10 μL of 10 μM target-specific

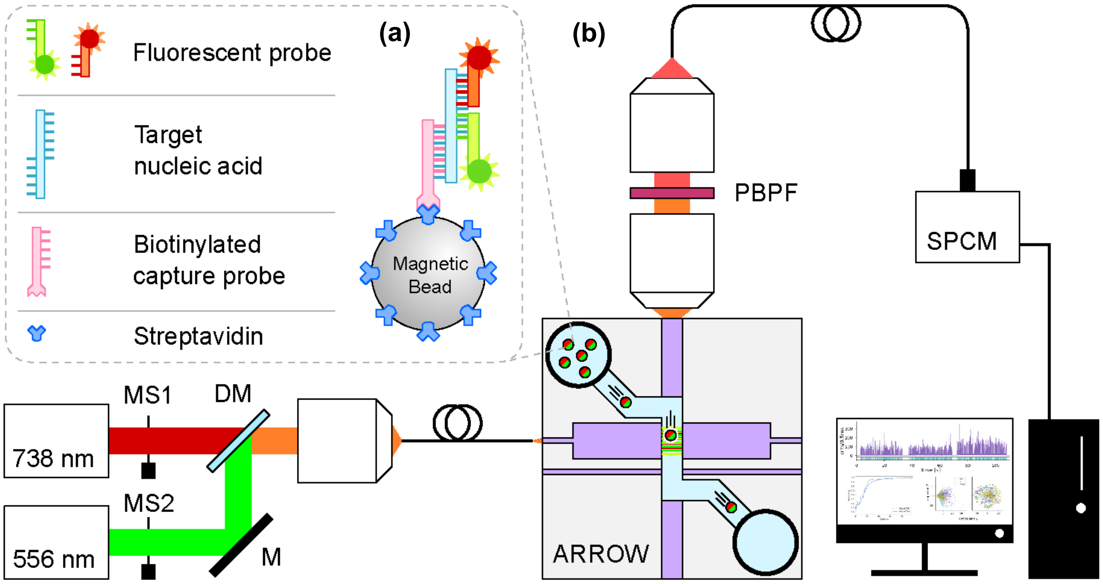


Figure 4.2: Experimental setup for multiplexed signals from Klebsiella Pneumonia.

fluorescent probe oligomers obtained from IDT DNA Inc. The resulting mixture is brought up to a final volume of 30 μL with 1XT50 buffer and incubated at 95 °C for 5 minutes, followed by 3 hours of incubation. Table 4.1 shows the sequences of the oligomers used for tagging the Klebsiella pneumoniae synthetic nucleic acid and binding to biotinylated capture probes.

Streptavidin-coated magnetic beads with a diameter of 1 μm, functionalized with target-specific biotinylated capture oligomers, are then mixed with the hybridized target-probe structure in a rotator mixer for one hour. The magnetic beads with the captured target-probe complex are then pulled down using a magnet and any unbound nucleic acid strands are washed off, after which the beads are resuspended. The final assay structure is shown in Fig. 4.2a (inset), and it has been previously demonstrated

Target name	Capture oligomer	Fluorescent probe oligomer	Color code
Klebsiella pneumoniae	/5BiotinTEG/ GTCGCCAGGC- CGCTG- GCGCGCTTG- GTCATAAAGT- TATCG- GTCTGGGCAGA3'	/5Alex546N/ CGCCGCCGAATTCCGGGAA- CATATCGGTCCAG3' 5'CGTACAGGGCGC- CAAGGTTACGACCG- TAGTC/3AlexF750N/	Green-Red

Table 4.1: Oligomers used for fluorescent tagging of K. pneumoniae.

that this labeling technique is highly specific to the target sequence [33].

Figure 4.2b reviews the fluorescence single-molecule detection setup based on ARROW (Anti-Resonant Reflecting Optical Waveguides) technology discussed in Chapter 3. The setup employs two lasers, operating at 738 nm (Ti: Sapphire, Del Mar Photonics) and 556 nm (SSD Nd:YAG, Shanghai Dream Laser Technology Co.), which are coupled into a single-mode optical fiber (F-SA, Newport) via a 60× microscope objective (Newport). Two modified PC cooling fans serve as mechanical shutters (MS1, MS2) to switch the optical paths for each color. The optofluidic chip is fixed on a custom 3D-printed stage using double-sided tape, and two brass cylindrical fluid reservoirs are attached to the liquid channel ends with wax. Negative pressure for sample flow inside the ARROW chip is provided by connecting the vacuum line to the outlet reservoir. The fluorescence signal from labeled targets is guided through the collection waveguide and collected from the side-facet using a 60× objective (Newport). The excitation light is eliminated by a penta-bandpass optical filter (FF01-440/521/607/694/809-25, Semrock) before coupling the collected light with a multi-mode fiber optic patch cable with an FC connector. A single-photon counting module (SPCM-AQRH, Excelitas Technologies) converts fluorescence photons into electrical pulses, and a time-correlated single-photon counting (TCSPC) board (TimeHarp 260 nano, PicoQuant) records time-tagged photon events onto the computer storage disk. The experiment is done in three stages, first with only 738 nm excitation (MS1=Open, MS2=Closed), second with only 556 nm (MS1=Closed, MS2=Open), and third with both 738 nm and 556 nm excitations active simultaneously (MS1=Open, MS2=Open). The sample containing the fluorescent targets is introduced into the chip by applying negative pressure to one of the reservoirs and then introducing the sample into the other reservoir using a pipette. The recorded fluorescence time trace from the chip is stored on a desktop computer (Fig. 4.2b). The multi-mode interference excitation waveguide produces different spot patterns for the two wavelengths in the intersecting fluidic channel with the number of spots N determined by equation (4.1)

$$N \times \lambda = \frac{n_c w^2}{L} \tag{4.1}$$

Where λ is the respective input wavelength, w is the effective waveguide width, n_c is the refractive index n_c, and L is the length of the MMI waveguide [4]. These spatial excitation patterns are transformed into the time domain as particles flow through the channel at a given velocity and fluorescence. Events from the first stage have six peaks, events from the second stage have eight peaks, and events from the third stage contain both 6 and 8 peaks. We named these events by splitting the word KPneu into two parts (“KPn” for six peaks and “eu” for eight peaks) to represent which probe has

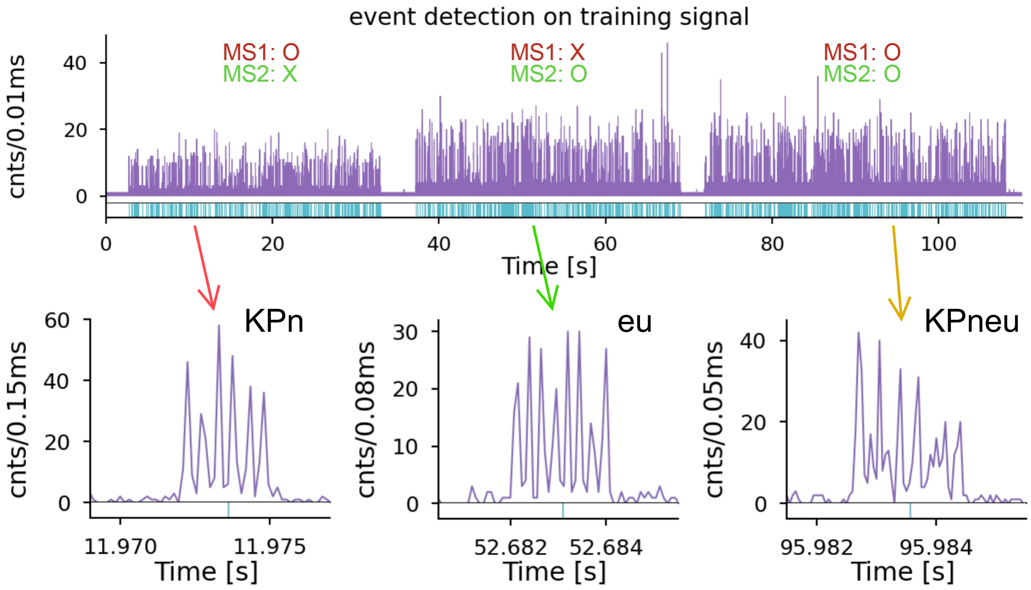


Figure 4.3: Fluorescence signal and example events from KPn, eu, and $KPneu$.

been excited in the detected event and will use these labels throughout the chapter. This approach effectively creates a 3× multiplex assay with three distinct signal shapes corresponding to the excitation wavelength selections.

Fig. 4.3 shows example events for each stage which illustrate the different fluorescence patterns and the nonidealities present in this configuration. These include finite background signal, peak-to-peak amplitude variations due to fabrication imperfections, and different peak-to-peak spacing Δt due to velocity variations from pressure fluctuations and position in the channel. A coefficient of variation of 44.85% is observed in the velocity distribution of 1,544 detected particles. Minimizing these nonidealities would require raising the production cost to increase the precision of the microfabrication processes. Some nonidealities, such as cross-section fluid flow speed variations, are

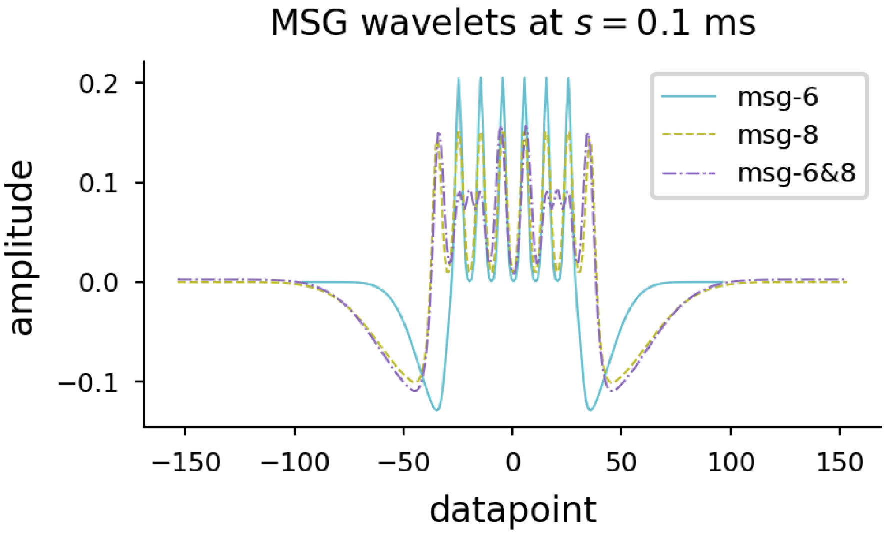


Figure 4.4: Wavelets used to detect events. MSG wavelets are used to detect and identify 6,8, and 6&8 peak signals.

simply due to the nature of the microfluidic channels being used. Point-of-care devices, meant to be low-cost and compact, must always deal with signal imperfections. A natural strategy is, therefore, to adapt to these imperfections by employing machine learning to recognize the signal pattern for a given device and detect and identify fluorescence events with good sensitivity and accuracy.

To begin our process of identifying signals, we first detect events from the fluorescence signal that has been recorded. In order to accomplish this, we employ our multi-scale event detection algorithm PCWA, which was introduced in Chapter 3. This algorithm utilizes multi-spot-Gaussian (MSG) basis functions to match the signals associated with events at each stage (as shown in Figure 4.4). The time location of detected events reveals which stage the event was recorded at so we can easily label them accordingly.

4.2.2 Preprocessing and polishing

Once events have been located within the time trace, a collection of cropped windows comprising 1024 data points centered on the time location of each event is generated to create a dataset for neural network analysis and classification. Events are then rescaled between 0 and 1 prior to feeding them into the classification model.

This paragraph describes a process for performing a multi-factor signal quality check. It presents an example of two overlapping KPn events and shows the extracted Δt parameter from the PCWA analysis on the left graph. On the right graph, a normalized cumulative summation of a down-sampled version of the signal on the left is shown, and a 3-segment piecewise function is fitted to it using a DE optimizer. The W and H parameters are annotated and extracted for quality check purposes. In (b), scatter plots of the extracted quality metrics demonstrate how outliers are rejected from the training dataset by setting threshold lines.

To ensure the accuracy of our neural network model's training, we remove any falsely identified or overlapping events from the dataset using a multi-factor signal quality check (see Fig. 4.5). The entire process of event detection, cropping, filtering, and annotation is automated, and does not require any user intervention. Fig. 4.6 shows a shuffled subset of events from the created dataset.

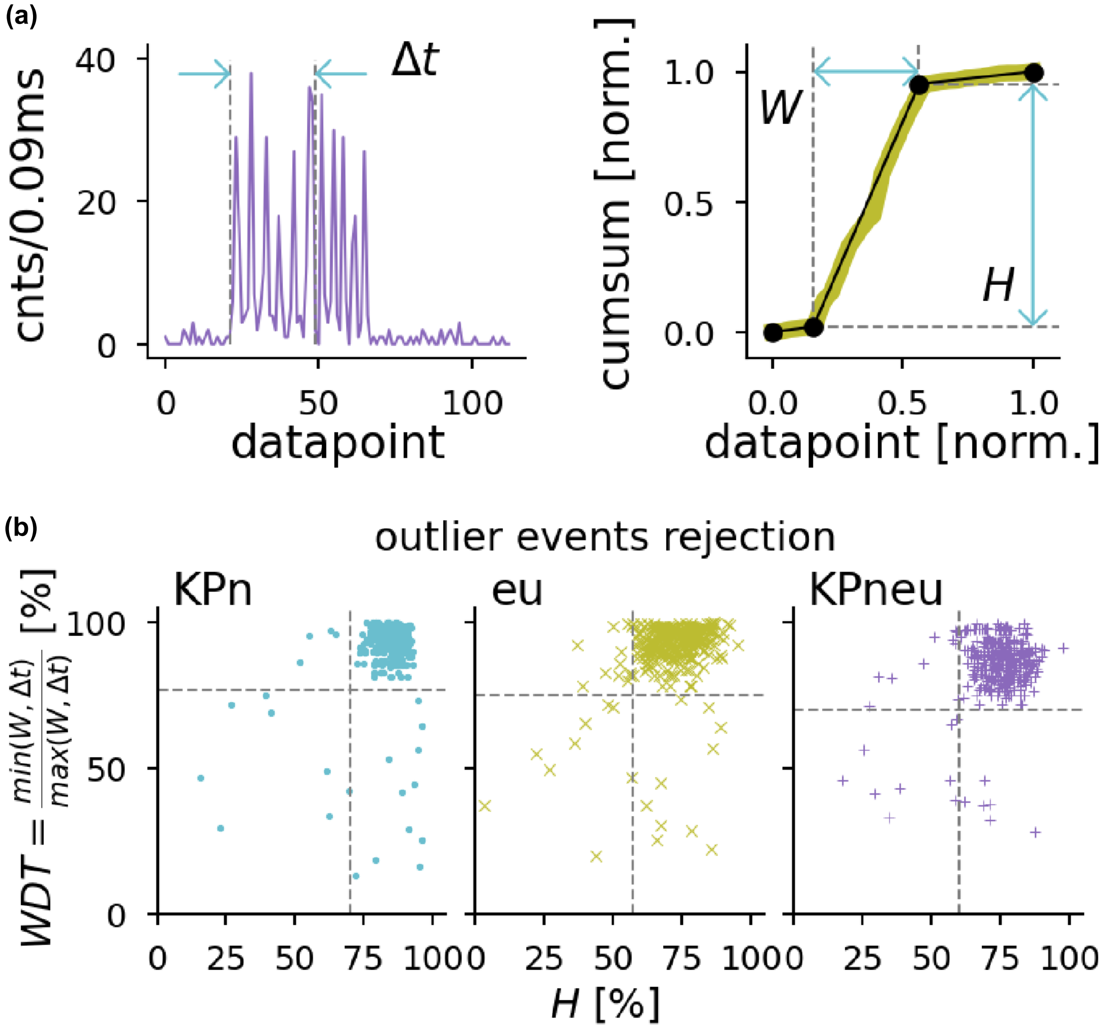


Figure 4.5: Multi-factor signal quality check to polish the dataset.

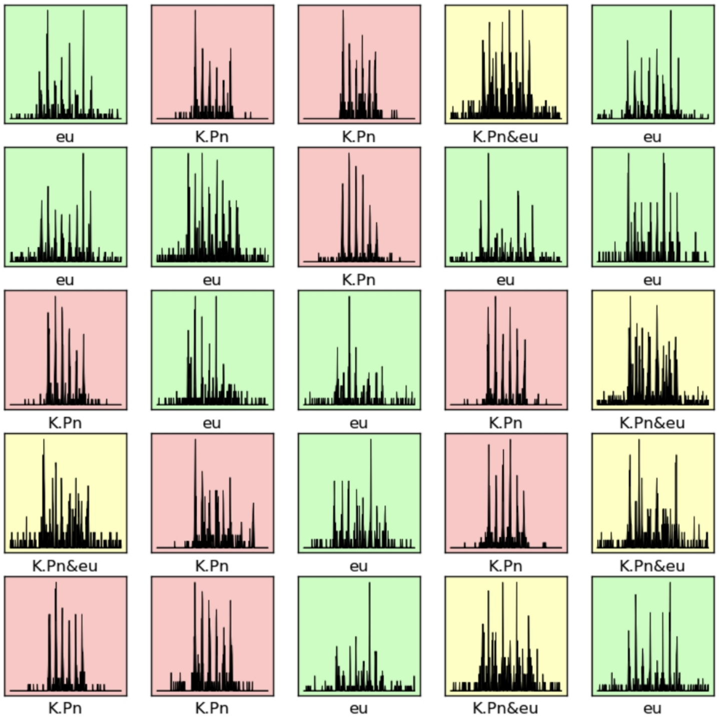


Figure 4.6: Examples from the labeled dataset.

4.3 Classification

Once we have created a labeled dataset, we can train a classification model and apply it to real tests. The entire dataset is usually divided into a train and test set with a split ratio of 20% (20% for testing and 80% for training). To get into machine learning, we start exploring neural networks (NN) by building a multi-layer perceptron (MLP) and then move to the convolutional neural network (CNN). The compatibility of NNs with AI accelerators is a special interest for the internet-of-things (IoT) and inferencing on the edge.

4.3.1 Multilayer perceptron (MLP)

Inspired by the brain, multilayer perceptrons are the very basic form of neural networks where a stack of connected layers is used to map inputs to the outputs in a feedforward fashion. Each layer consists of neurons, biases, and an activation function which can be described as

$$a_j^l = \sigma(\sum_k w_{jk}^l a_k^{l-1} + b_j^l) \tag{4.2}$$

l, j are the layer number and neuron number in layer l respectively. a is the value of neurons connected to the next layer's neurons with connection weight w. Each layer has an optional bias value b and a linear or non-linear activation function σ. Fig. 4.7 visualizes an MLP structure with input and output layers and a hidden layer in between. Values of weights and neurons in the hidden layer(s) are randomly assigned

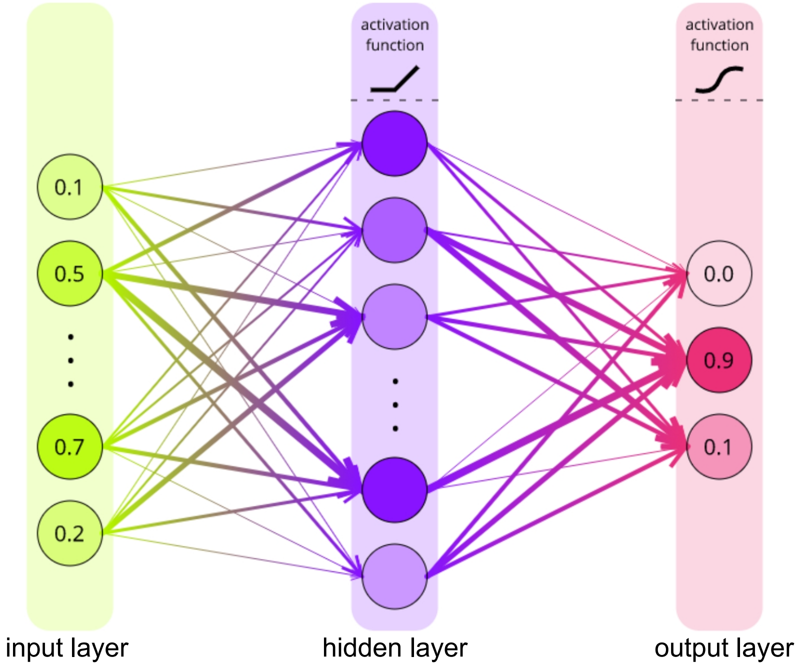


Figure 4.7: A simple MLP architecture.

at the beginning and during a process called training, these values converge to optimal values.

4.3.1.1 Designing and training an MLP model

Here, we demonstrate how to create and train an NN model using Python and TensorFlow. We define a very small and simple MLP using a 'sequential' model from Keras backend (inside TensorFlow):

```
neural_net = keras.Sequential([
  keras.layers.InputLayer(input_shape=(180,)),
  keras.layers.Dense(128, activation='relu'),
  keras.layers.Dense(3, activation='softmax')
])
```

This snippet defines an MLP neural network consisting of a single hidden layer of 128 neurons and a ‘relu’ activation function. The input shape is determined by the events data shape (here, 180 data points) and the output size is determined by the number of classes we want to identify (here, 3). The output layer uses the ‘softmax’ activation function described as

$$
\begin{gathered}
\sigma(x) = x^{+} = max(0, x), \text{ ‘relu’} \\
\sigma(x_c) = \frac{e^{x_c}}{\sum_{i=0}^{k} e^{x_i}}, \; for\ c = 0, ..., k, \text{ ‘softmax’}
\end{gathered}
\tag{4.3}
$$

The model is then compiled with the optimizer, loss function, and tracking metrics for later evaluation. The Python snippet below shows how to compile our neural network model

```
neural_net.compile(
  optimizer='adam',
  loss='sparse_categorical_crossentropy',
  metrics=['accuracy']
)
```

We use the ‘Adam’ optimizer which is named from adaptive moment estimation as a popular optimizer for gradient-based optimization of stochastic objective functions [82]. The optimizer passes training data through the model in a process called forward propagation and calculates the output. The output from the NN is then compared to the true values (labels) via a loss function, and then this error propagates backward through the network in a backpropagation process. Backpropagation utilizes the derivatives of activation functions to adjust and update weights and neuron values of hidden layers.

This is done in TensorFlow by simply calling the fit() method. We pass training data x, and y (labels) plus a number of iterations (epochs) we want the neural network to get exposed to these data.

```
neural_net.fit(x_train, y_train, epochs=50)
```

Once the training step is completed, loss and other metrics (i.e. accuracy) can be plotted versus the epochs to check how training was done. Fig. 4.8 includes a few examples of training plots for the number of neurons of 10, 100, and 1000 for the hidden layer. Here, the number of neurons is called a hyperparameter and there are usually several hyperparameters that can be optimized for a NN model. Obviously, the higher number of neurons increases the capacity of the neural network and can fit better to the training dataset, however, this will lead to an increase in model size, longer training time, slower inferencing, and the possibility of overfitting.

Dropout is a regularization technique to avoid overfitting and prevent neural networks from getting optimized to local maxima instead of global maxima [83]. We notice that adding a dropout layer as below can significantly improve the robustness of the model.

```
neural_net = keras.Sequential([
  keras.layers.InputLayer(input_shape=(180,)),
  keras.layers.Dense(128, activation='relu'),
  keras.layers.Dropout(0.2),
  keras.layers.Dense(3, activation='softmax')
])
```

There are two curves per color, solid and dashed for training and validation

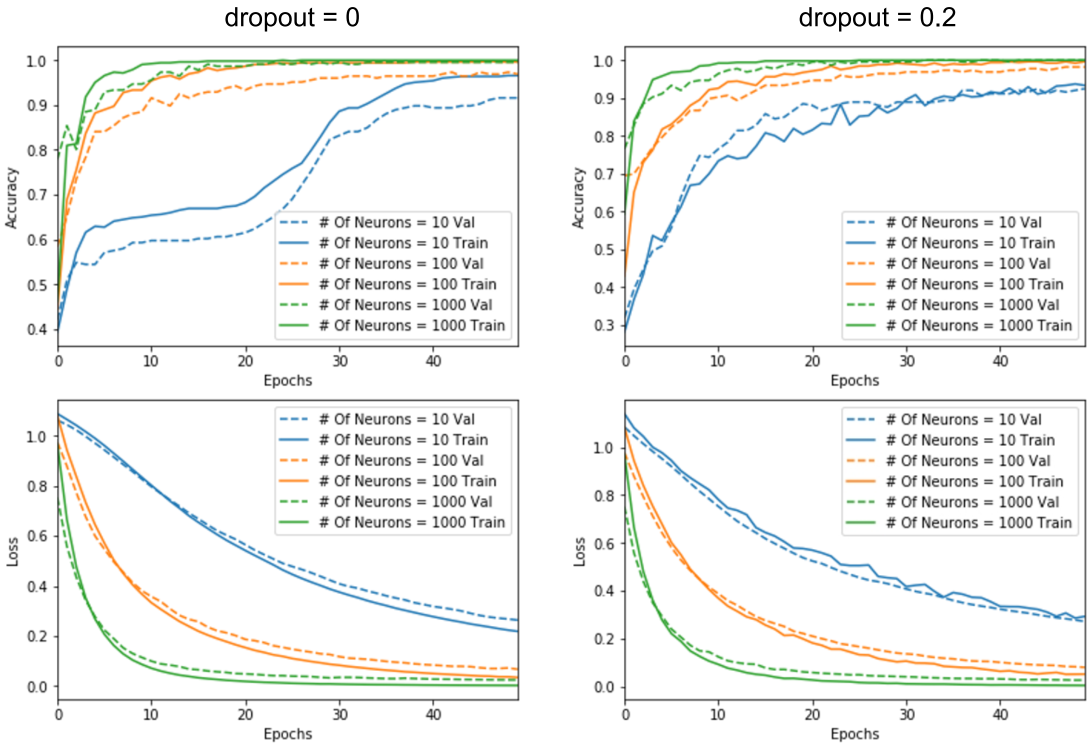


Figure 4.8: Training of MLP models and hyperparameter.

data respectively. The validation curve is a portion of training data held by the optimizer for validating the model at every epoch. This is different from the final test set because the neural network gets exposed to validation data during training, whereas test data will be kept unseen.

In order to test and evaluate the trained model with unseen test data, we use the following snippets

```
pred = neural_net.predict(x_test)
```

predict() method returns the output values of the MLP network, while evaluate() takes additional y_test value to return loss and accuracy.

```
test_loss, test_acc = neural_net.evaluate(x_test, y_test)
```

We showed, step-by-step, how to implement an MLP for our dataset and even though the results are great, the MLP models suffer from translational and temporal stretch errors in the data. This problem is a major issue because the nonidealities in multi-spot signals result in inaccurate localization of events in the time domain. Also, target particles flowing at different velocities result in stretched or compressed signals in the time axis, thus leading us to explore a powerful and better-suited architecture of neural networks called convolutional neural networks (CNN).

4.3.2 Convolutional neural network (CNN)

The main difference between MLP and CNN is the filters. CNN utilizes layer(s) of filters (kernels) for the convolutional operation and these filters get trained during

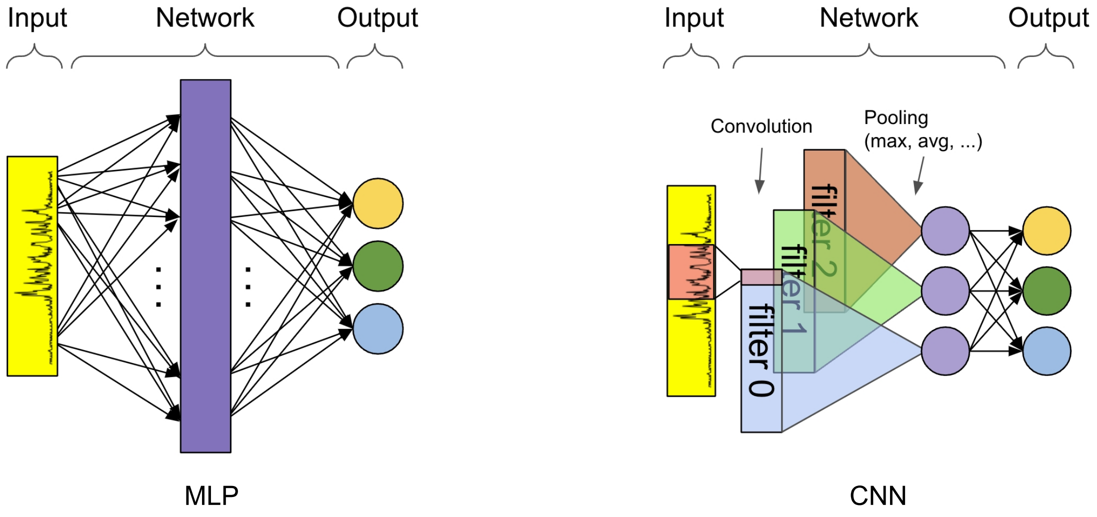


Figure 4.9: MLP vs. 1D CNN layers.

the training step to match with specific features of the input signal. For example, one filter will extract positive slopes, the other filter will detect peaks, and so on. The other approach can be using bigger filters (length comparable to the input signal) and trying to fit filters to the entire signal, similar to what we did in Chapter 3.

A pooling (max or average) layer is usually followed after convolutional to keep only the enhanced part of the convolution result, reducing the internal layers' size. Multiple layers of convolution+pooling can be stacked to go deeper and draw out high-level features. Once features are extracted by the convolutional layer, a fully connected classifier is used to map these features into the output classes (similar to MLP, see Fig. 4.9). There are some benefits to using a convolutional layer, the first is gaining some tolerance to translational variations, the second is a reduction in the size of parameters, and the third is faster training.

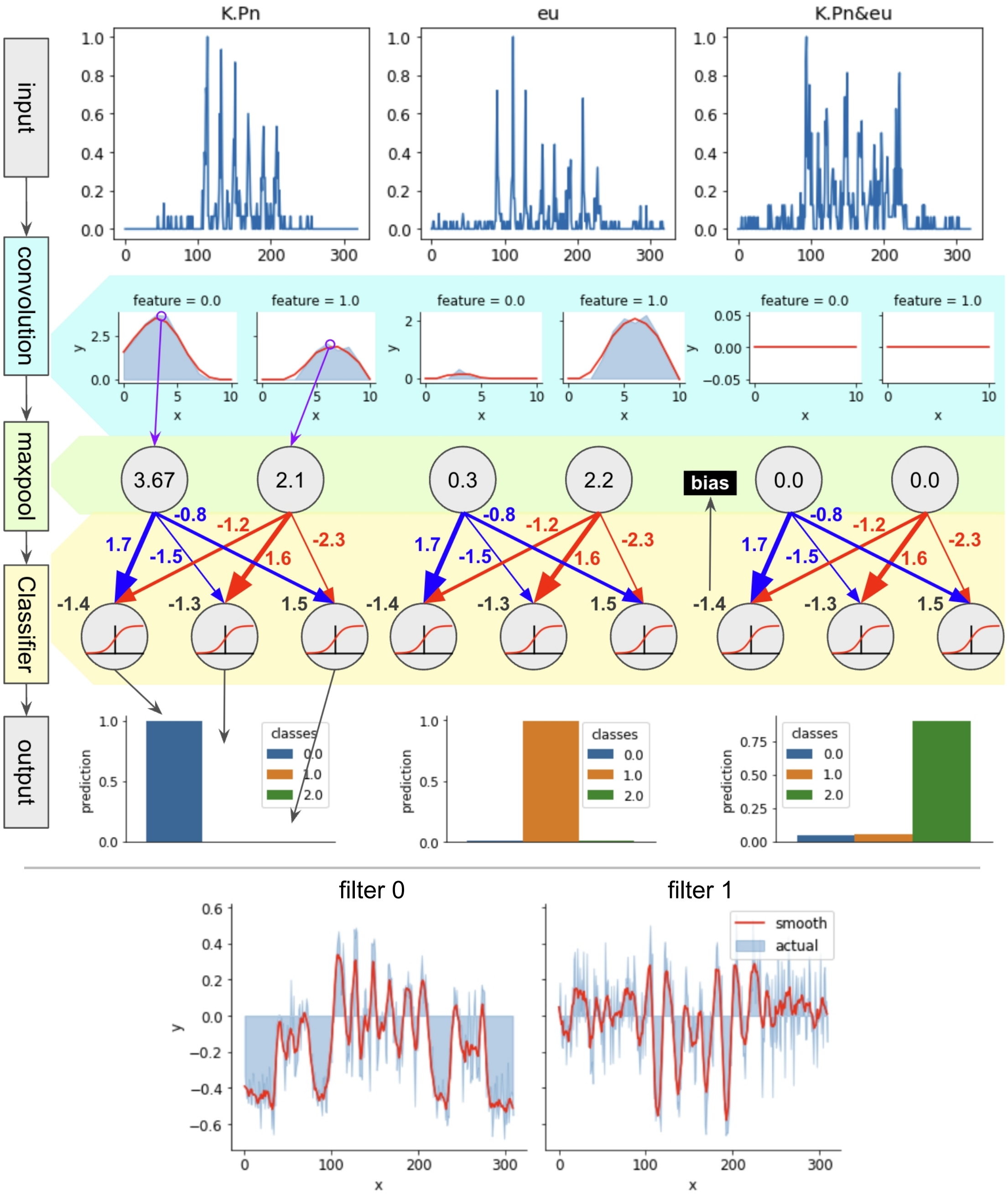


Figure 4.10: 1D CNN classification model working principle.

As a comparison, to achieve an accuracy of over 99% with a single hidden layer MLP model, we use 500 nodes for the hidden layer and a total of 162k trainable parameters is required. On the other hand, a CNN model with 6 filters with a length of 310 is capable of achieving similar performance with only $\sim$ 2k trainable parameters. This type of 1D-CNN model can be visualized as shown in Fig. 4.10.

The input signals to the CNN model can be a spectrogram instead of the raw 1D signal to include temporal and spectral information in a 2D image. This tends to be more interesting when the distribution of velocity for events becomes broader. A concept of a multi-layer CNN model where deeper features are conveyed is illustrated in Fig. 4.11. A DNN model consisting of multiple convolutional layers and a fully connected (dense) classification layer at the end. Based on PCWA analysis, most of the MSG signals collected from labeled targets sit in a 0.1 ms $< \Delta t <$ 1.0 ms scale range. A window size of 1024×0.01 ms $= 10.24$ ms is big enough to enclose even very slow-moving particles with the number of peaks (N) below 10 (1.0 ms $\times N < 10.24$ ms). The input signals are first converted into a 128×128 spectrogram using a short-time Fourier transform (STFT) and then fed into layers of a 2D convolutional network. In the end, a fully connected layer uses the extracted features from filters for classification purposes. A summary of the layers used in the DNN model for event classification is presented in Table 4.2.

This model is relatively small and takes less than an hour to train on a desktop machine with Nvidia Quadro P1000 GPU. The training was stopped after 100 epochs to avoid overfitting. Fig. 4.12 plots the metrics of the model over the training epochs.

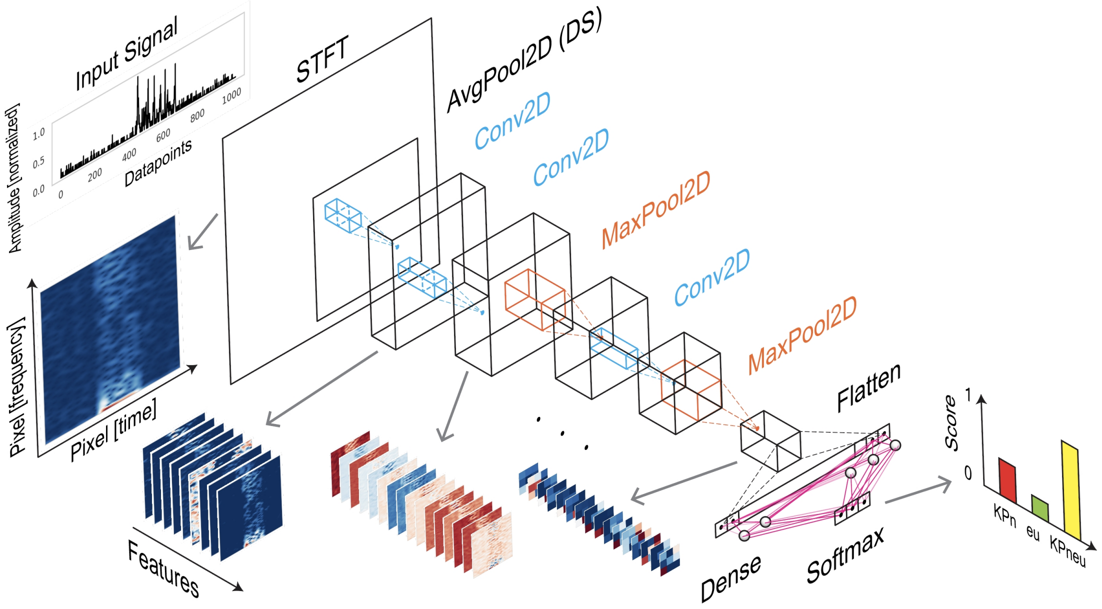


Figure 4.11: DNN model for event classification.

Layer type	Output shape	Param #
Input	(None,128,128,1)	0
AveragePooling2D	(None,64,64,1)	0
Conv2D	(None,62,62,9)	90
BatchNormalization	(None,62,62,9)	36
MaxPooling2D	(None,20,20,9)	0
Dropout	(None,20,20,9)	0
Conv2D	(None,16,16,25)	5650
MaxPooling2D	(None,3,3,25)	0
Dropout	(None,3,3,25)	0
Flatten	(None,225)	0
Dense	(None,3)	678

Total params: 6,454
Trainable params: 6,436
Non-trainable params: 18

Table 4.2: Summary of DNN model layers for event classification.

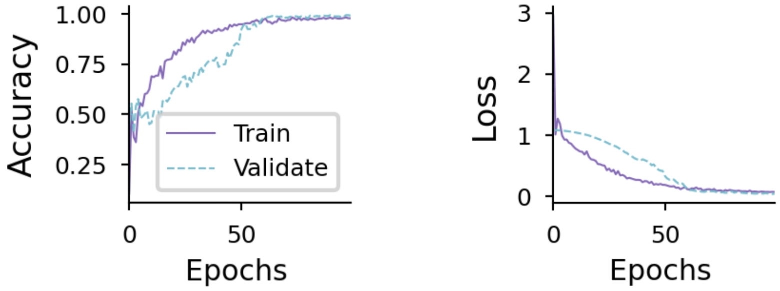


Figure 4.12: Training progress for DNN model.

A categorical cross-entropy loss function L defined by Eq. 4.4 is minimized by an Adam optimizer and the loss value drops below 0.1 after 90 epochs. The final output layer with a softmax function ($k = 2$) as described in Eq. 4.5 returns the probability of three possible classes (p_c) for a given event.

$$L = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{c=0}^{2} y_{i,c} log(p_{i,c}) \tag{4.4}$$

$$p_c = \sigma(x_c) = \frac{e^{x_c}}{\sum_{i=0}^{2} e^{x_i}},\ c = 0, 1, 2 \tag{4.5}$$

In this experiment, we use events detected from $\sim$ 250 nL of the sample with an approximate concentration of 10^6/mL which is about 300 events/class to train the model. This was enough to get the model converged accuracy of over 99%.

4.4 Transfer learning and detection on the edge

With the rise of the Internet of Things (IoT) and edge technology [84], the demand for powerful devices at the edge is more crucial. The edge users are usually

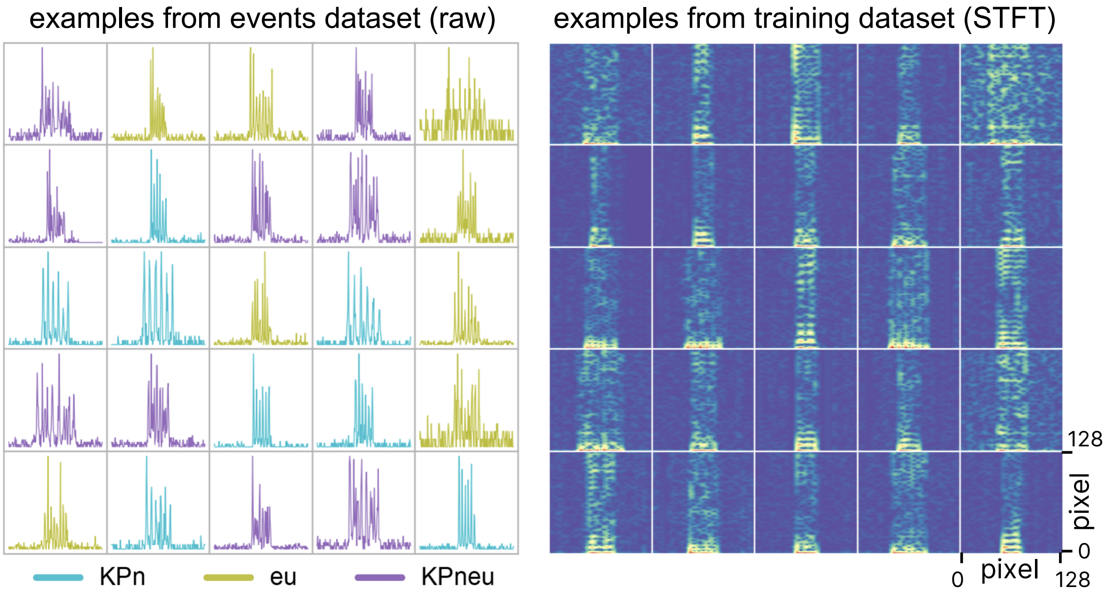


Figure 4.13: Examples from the training dataset.

end-users and the traditional way of communication with a cloud-based server is gradually moving towards end-user side processing of data for many reasons such as limited network access in a certain area, bandwidth usage, privacy, etc [84]. Here, we explore how we can stretch the current pathogen detection platform toward the new growing edge technology. For this purpose, an AI-specific edge computing device, Google Coral Dev board [85], is selected for a few reasons: it is relatively cheap ($140 by the time of this study), small footprint (Raspberry Pi size), power efficient (2 TOPS/watt for Edge-TPU), integrated ARM processor and running a linux-based operating system. Also, the availability of libraries and resources to develop custom models was another important factor to choose the Coral Dev board over other options. Table 4.3 summarizes some of the important specifications for the Coral Dev board:

Edge TPU System-on-Module (SoM)
- NXP i.MX 8M SoC (Quad-core Arm Cortex-A53, plus Cortex-M4F)
- Google Edge TPU ML accelerator coprocessor
Memory
- 1 or 4 GB LPDDR4 SDRAM (4-channel, 32-bit bus width)
- 8 or 16 GB NAND eMMC flash memory
Network
- 10/100/1000 Mbps Ethernet/IEEE 802.3 networks
- Wi-Fi 2x2 MIMO (802.11a/b/g/n/ac 2.4/5GHz)
- Bluetooth 4.2
Baseboard
- 40-pin I/O header
- Gigabit Ethernet
- HDMI 2.0a (full size)
- MicroSD slot
Supports Mendel Linux (a derivative of Debian)

Table 4.3: Coral Dev Board features.

The compatibility of TensorFlow with Coral board makes the transfer of the trained model from the desktop machine to the Edge-TPU device an easy process. The original TF model is based on 32-bit floating point digits, whereas Edge-TPU works with 8-bit integer numbers. One of the steps done in transfer learning is to convert models and operations to compatible data types and operations. There is a list of supported layers and operations by Edge-TPU accelerator [86] and we make sure that our DNN model is built by only these supported elements. The rest of the framework and if there is any unsupported operation/data type by Edge-TPU, will run on the main CPU (ARM Cortex-A53). Fig. 4.14 illustrates the key blocks of the developed real-time edge inferencing framework with arrows indicating the data flow direction.

The stream of incoming APD (sensory) data gets stored in a memory-mapped

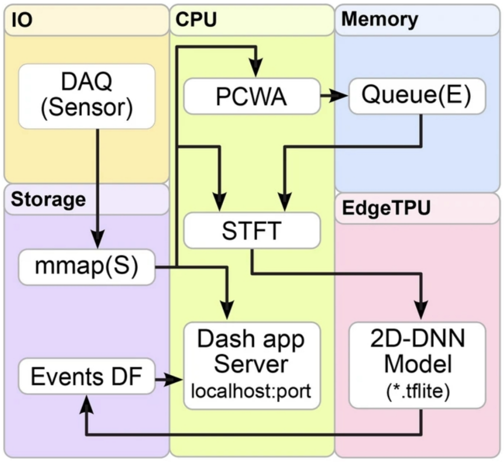


Figure 4.14: Real-time edge inferencing framework.

file in the storage device (MicroSD card). This data file then is accessed by the CPU for two purposes: stream out to the user for visualization and to the event detection and identification process for analysis. The visualization is done by running a Plotly Dash app in a separate process to take advantage of multiprocessing for minimal delay. In parallel, another process runs the PCWA code on chunks of input data to detect events using a non-specific general square wavelet (see Fig 4.15). Here, the wavelet will match best with the entire multi-peak signal without extracting any information about individual peak spacing or the number of peaks.

After locating events with PCWA, a cropped window centered at the center of the event is pushed into the events queue. The queue is a first-in-first-out (FIFO) queue and it is used to help with synchronizing multiple processes accessing a shared list. Here, for instance, a separate process running in parallel to PCWA, takes cropped events from the queue, calculates STFT and runs inference on the Edge-TPU unit.

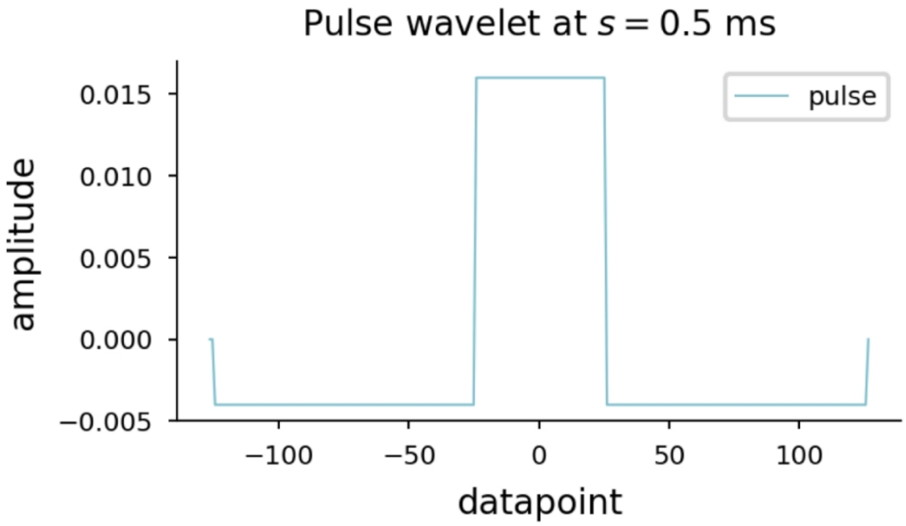
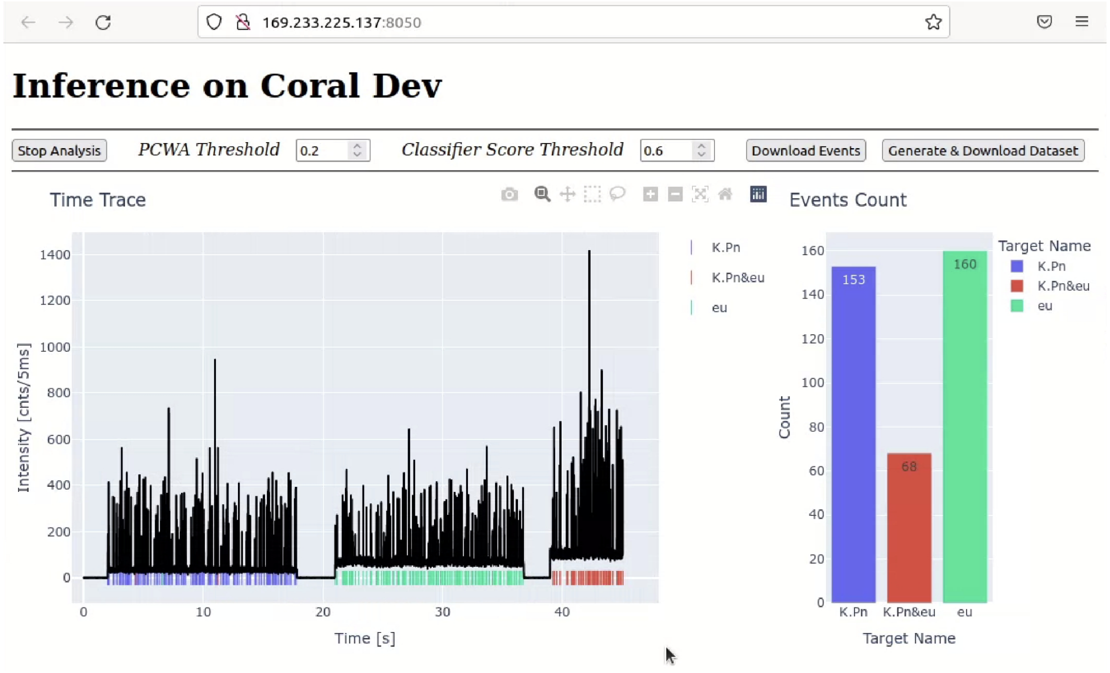


Figure 4.15: Square wavelet used for multi-peak event detection.

This prevents any possible propagation delay from one process to another process and they can generate/take data from the queue in an asynchronous fashion. The result of the inference on Edge-TPU (identified class of the event) is then stored in the events dataset for storage and visualization purposes. The Dash app visualizes both the raw input signal and identified events as colored markers in a dashboard in real-time. Dash app runs as a server and a user can access the dashboard on any web browser by pointing to the Coral IP address and port number (here, 169.233.225.137:8050) specified in the Dash app configurations (see Fig. 4.16). The entire framework is implemented in Python 3 and runs locally on the Coral Dev board. After manual inspection of detected events, we found over 99% of events getting picked correctly by the PCWA algorithm.

As mentioned, the PCWA takes input data stream in smaller chunks (1 s) and we found this a reasonably fast way of updating results every second in the dash. This,



Inference on Coral Dev

PCWA Threshold 0.2

Classifier Score Threshold 0.6

Figure 4.16: Real-time Dashboard developed in Plotly's Dash.

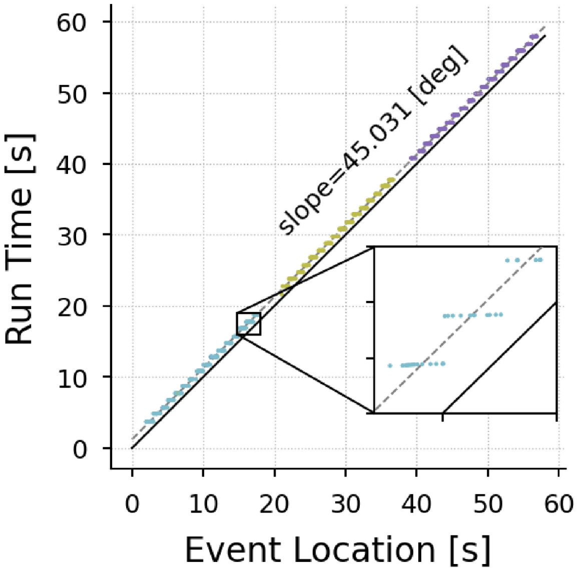


Figure 4.17: Real-time performance running on Coral Dev.

however, adds an overhead delay of ∼ 1 s, but since the data flow in the framework is pipelined and runs in parallel processes, there is no accumulation of delay. The ∼ 45° slope of events detection time versus the actual events time location in APD data trace reflects the real-time performance achievable in a compact, cheap, low-power edge device (Fig. 4.17).

Finally, we compare the accuracy of the developed model and compare it to our previously used shift-and-multiply (SaM) event identification algorithm [87] both on a desktop machine and on the edge device with a quantized version of the model. As shown in Fig. 4.18, the DNN model is capable of forming linearly separable clusters when tested on the test dataset. On the other side, SaM suffers from the poor classification of KPneu events. A minor drop in performance is visible in the quantized model which

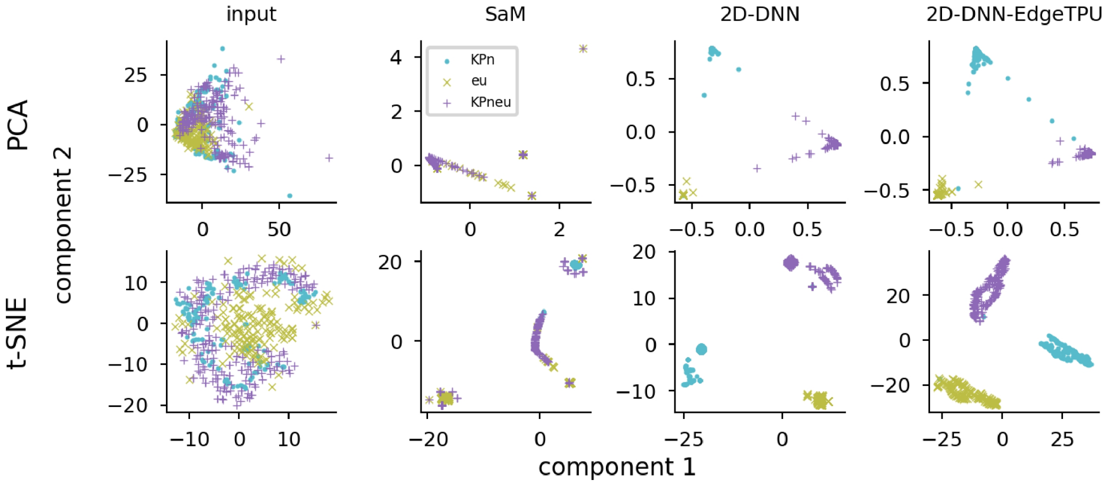


Figure 4.18: Classification performance.

is inevitable with a lower data precision (8-bit integer versus 32-bit floating). Principle component analysis (PCA [88, 89]) and t-distributed stochastic neighbor embedding (t-SNE [90]) are two of the most famous linear and non-linear dimensionality reduction techniques and are used here to help to visualize higher dimensional output data.

Confusion matrices and the receiver's operating characteristic curves (ROC) are shown in Fig. 4.19 and Fig. 4.20 respectively. As expected, DNN models (CPU and Edge-TPU) versions outperform the SaM in classification accuracy with over 40% higher area under the curve (AUC). Out of a total of 485 detected events by PCWA, 99.8% of them (484) are correctly classified by the 2D-DNN model and a combined accuracy of 99% is reported when undetected events are taken into account (488 manually inspected ground truth events).

The Edge-TPU accelerator chip in the Coral Dev board is optimized for inferencing and in this study, we found 2× faster inferencing when done on Edge-TPU

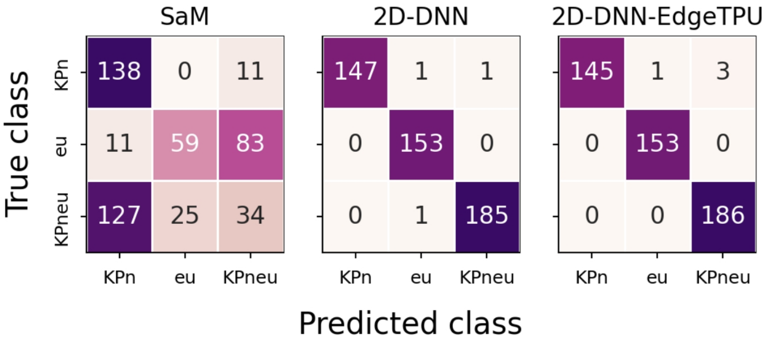


Figure 4.19: Confusion matrices for different classifiers.

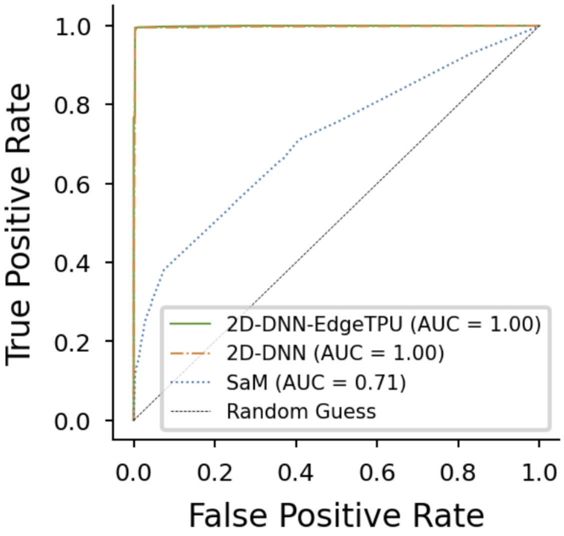


Figure 4.20: ROC curves for different classifiers.

compared to ARM CPU on the board. This is $\sim$2 ms inference time on average for each event with an event window length of $\sim$ 10 ms thus emphasizing the real-time performance.

Chapter 5

Time-wavelength division multiplexing (TWDM) for broad dynamic range

In this chapter, we will propose and investigate a novel experimental configuration and data analysis platform for high dynamic range detection of pathogens.

5.1 Detection dynamic range in ARROWs

The ultrasensitive ARROW biosensors as one of the successful optofluidic platforms have shown promising results in amplification-free fluorescence detection of biomolecules down to single molecule [33] with simultaneous multiplexed detection of upto 7× kinds of targets [36, 87]. These sensors are designed to detect and count individual target molecules (digital detection) and work only if there is no more than one target crossing the excitation volume. Fig. 5.1 illustrates two possible scenarios in an ARROW device depending on the sample concentration. The analog regime is where

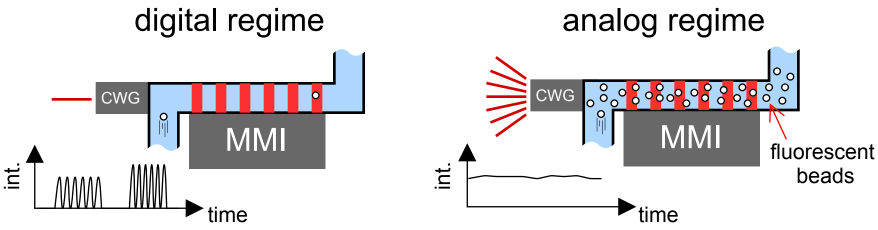


Figure 5.1: Digital and analog detection regimes.

multiple target particles get excited at different locations inside the excitation volume and create a superimposed (analog) signal.

If we have only one type of target flowing inside the channel (singleplex detection), there are two approaches to measuring the concentration:

- dilute the sample to work in the digital regime
- convert analog signals intensity to concentration

These solutions are very simple and require no additional change in the system, however, they will fail in multiplex detection. Diluting approach can be still used for multiplex detection if target molecules have similar concentrations. Diluting the sample is not very feasible because, for an unknown concentration of target molecules within a sample, we may have to do multiple steps of dilution + measurement until the time signal shows digital events. On the other hand, the analog signal doesn't contain any spectral or temporal information of target types, therefore, not useful for multiplex detection of two or more highly concatenated targets.

5.1.1 Concentration estimation: digital regime

We can calculate the digital concentration limit (upper limit) based on the chip dimensions and flow rate for a given ARROW chip. Fig. 5.2 illustrates an excitation volume inside an ARROW chip. Eq. 5.1 is used to calculate the concentration based on channel dimensions and flow rate calculated from the APD signal.

$$c = \frac{N}{Qt}[particles/m^3] = \frac{N}{\bar{v}h_{exc}dt}[particles/m^3] = \frac{N}{N_A\bar{v}h_{exc}dt \times 10^3}[M] \tag{5.1}$$

where N is the number of events in time window t, N_A is Avogadro's constant ($6.022 \times 10^{23}mol^{-1}$), and Q is the flow rate calculated using the average velocity of detected events, v, and the excitation volume dimensions (h_{exc}, d). h_{exc} is the full-width-half-maximum (FWHM) of the excitation mode and d is the length of the excitation spots (see Fig. 5.2). As the target concentration increases, we will eventually reach the point where multiple targets occupy the excitation region and individual events cannot be identified any longer. We can estimate the limit of the digital range $c_{D,max}$ from the excitation volume (V_{exc}) using Eq. 5.2

$$\begin{gathered} V_{exc} = dhw[m^3] = dhw \times 10^3[L], \\ c_{D,max} < \frac{1}{N_A V_{exc}[L]} \end{gathered} \tag{5.2}$$

The chip used in this research has a diameter (d) of 12 mm, a hexagonal chamber size (h_{exc}) of 3 mm, and a width (w) of 100 mm. The maximum fluorescence signal that can be obtained from this chip is $c_{D,max} = 462fM$. If the concentration

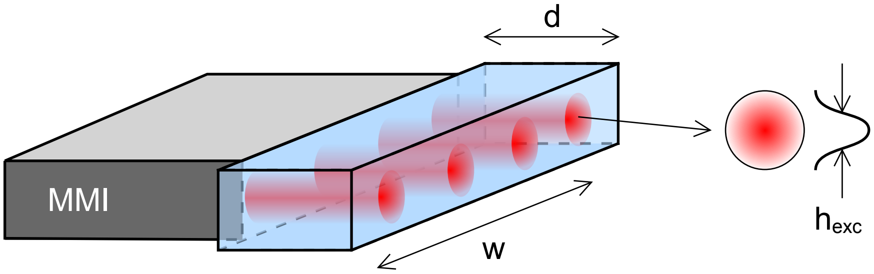


Figure 5.2: MMI excitation volume.

exceeds this limit, a continuous and analog fluorescence signal will be detected, the intensity of which will depend on the concentration.

5.2 Time division multiplexing (TWDM) concept

In this section, we propose a time-wavelength division multiplexing (TWDM) scheme to address the mentioned problems. First, we explore the concept by devising a simple experiment and minimal change in the current setup. The conventional single-molecule-detection (SMD) setup with two laser sources running at 633 nm (HeNe, Melles Griot) and 488 (LuxX® 488-60, Omicron-Laserage Laserprodukte GmbH) is shown in Fig. 5.3. The main change is the addition of two chopper blades to modulate the input lights.

By placing chopper blades at individual laser paths and running at different frequencies (6 Hz and 16 Hz), we are able to modulate input laser lights. The modulation is a square pulse modulation with a 100% peak-to-valley ratio and 50% pulse width (50% on cycle, 50% off cycle). Fig. 5.4 shows an APD time trace recorded from a time-division

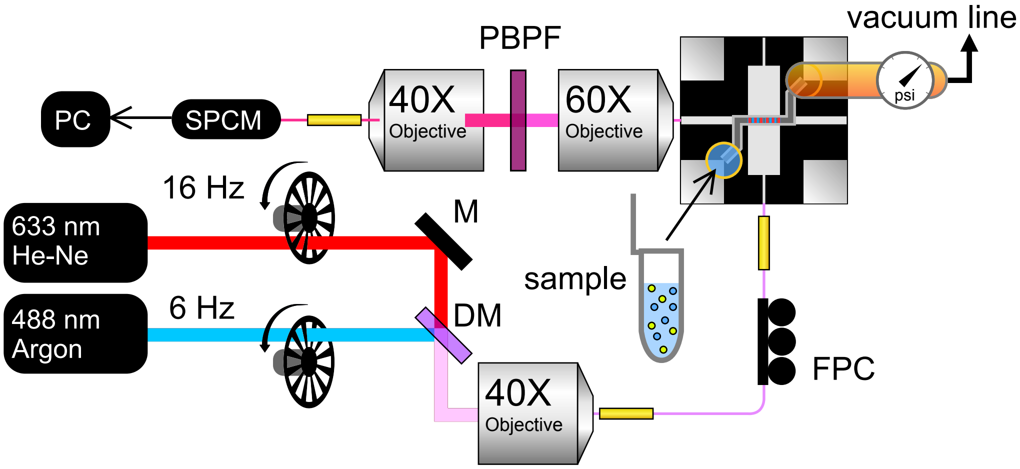


Figure 5.3: Modified setup for time-wavelength division multiplexing.

multiplexing setup with a mixture of 200 nm crimson and yellow-green fluorescent beads (FluoSphere™ carboxylate polystyrene beads, ThermoFisher Scientific). The solution is in a digital regime with a Concatenation of~ 83 fM (below $c_{D,max}$). Individual events are detected by running a PCWA algorithm with a complex Morlet wavelet and the individual events are shown as bright spots in the D-band. The scale range is divided into two regions, analog band (A-band) and digital band (D-band). Depending on which lasers are on, there are bright bands observed in A-band reflecting the chopper modulation frequencies.

By looking into the zoomed-in window of the signal, we notice digital events detectible in the 'on' states of the lasers and there are parts that are missing events due to 'off' states of the lasers. This is a big concern at the lower end of the concentration range where individual events should be detected to estimate concentration in a short

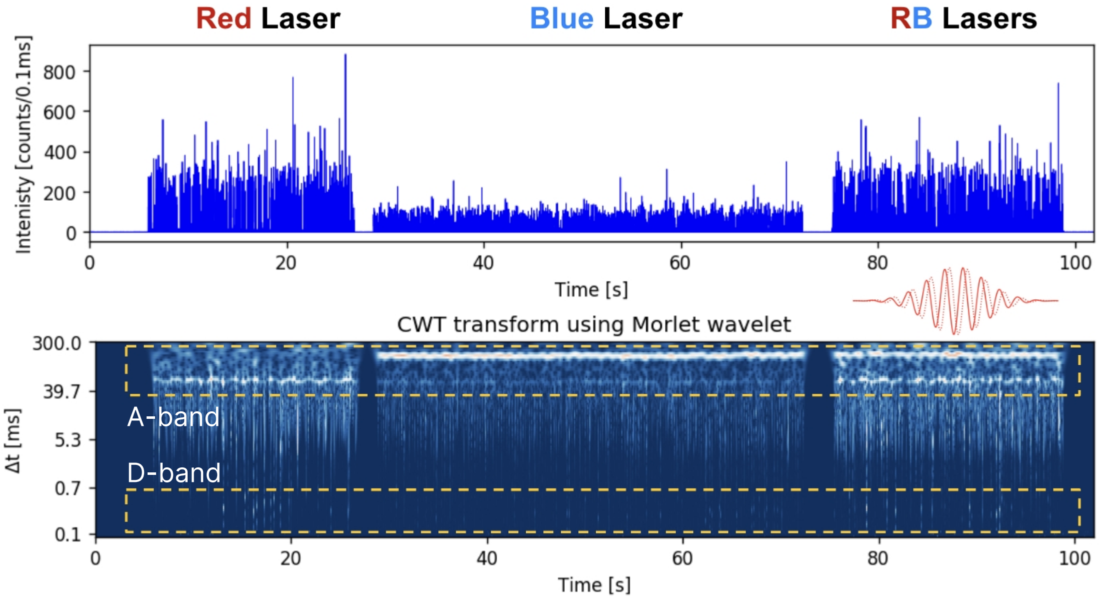


Figure 5.4: APD trace from the TWDM signal (chopper modulation) for the digital regime (83 fM).

time.

At the higher concentration working regime (analog regime), the overlapping signals form an analog signal similar to the modulated input pulses. Fig. 5.6 shows a similar APD trace recorded for a sample of 100× higher concentration at ∼ 8.3 pM (above $c_{D,max}$). Obviously, there is no event detectable (see Fig. 5.6 and Fig. 5.7) and the analog bands are much brighter. In the digital regime, most of the analog bands are coming from the background signal and random digital fluorescent events do not contribute to the bright bands at certain frequencies.

We confirm that the TWDM concept is capable to provide concentration-related information at higher concentrations beyond the digital detection limit. There are however certain problems and limitations with the chopper blades. The first and

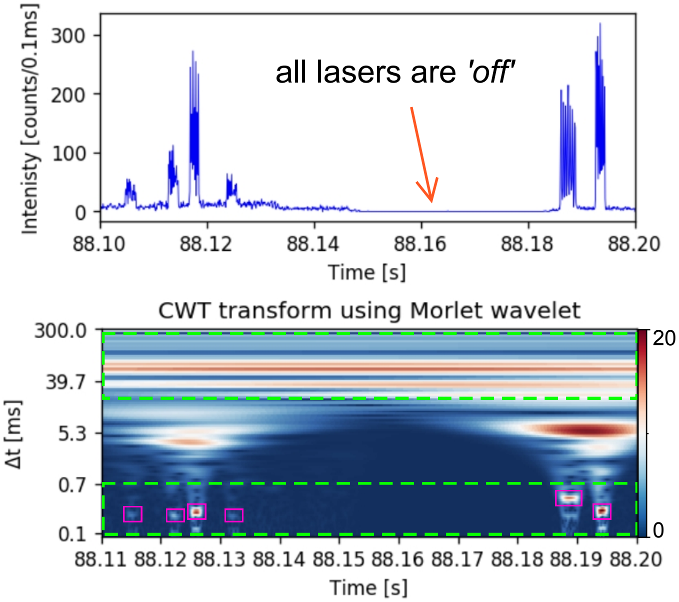


Figure 5.5: Zoomed-in view of APD trace from the time-division multiplexing signal (chopper modulation) for the digital regime (83 fM).

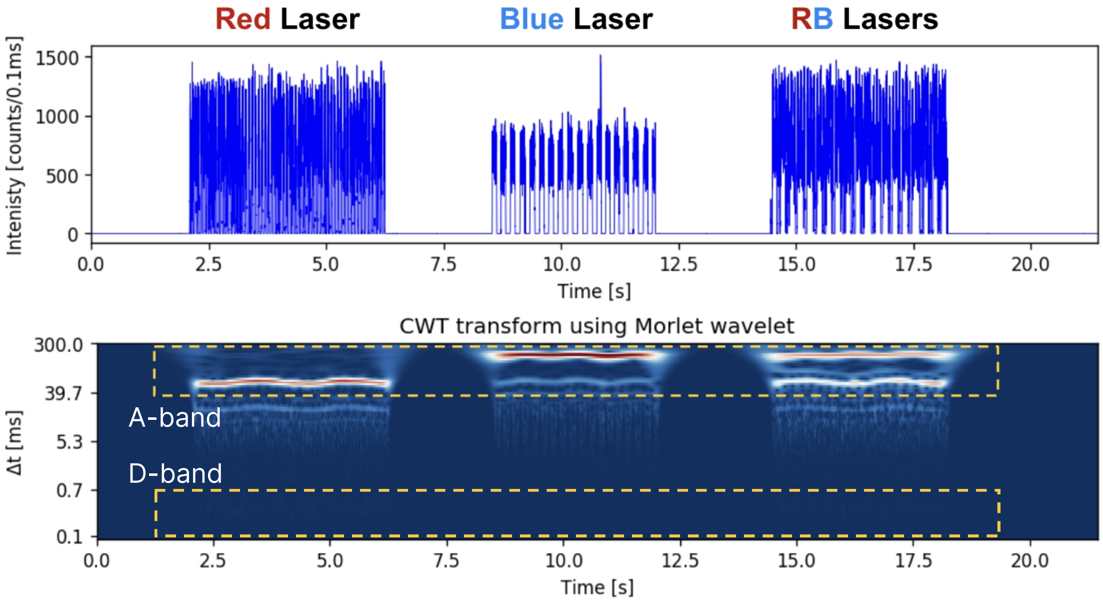


Figure 5.6: APD trace from the time-division multiplexing signal (chopper modulation) for the analog regime (8.3 pM).

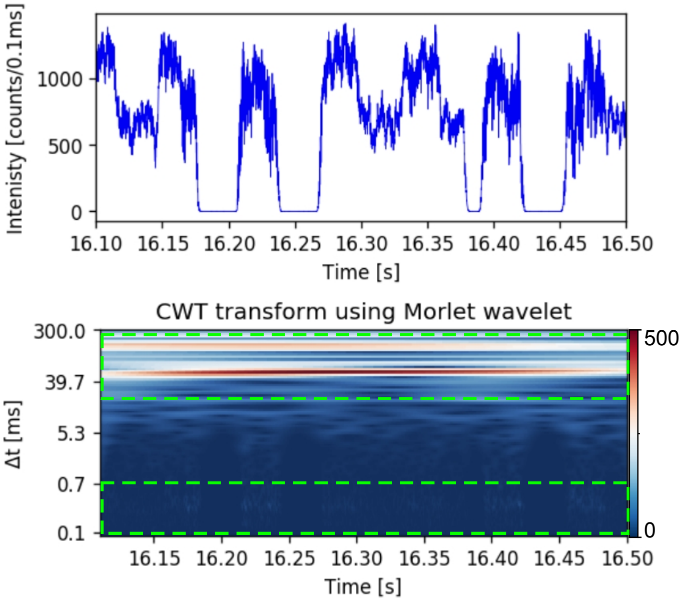


Figure 5.7: Zoomed-in view of APD trace from the time-division multiplexing signal (chopper modulation) for the analog regime (8.3 pM).

main one is the total blocking of beams and forming 'on' and 'off' states in which there will be missing events for the 'off' periods. The other problem is the fluctuation of chopper blades spinning frequencies due to mechanical fluctuations. In the next section, we will explore another modulation scheme based on the analog modulation of laser diodes to tackle this problem.

5.3 Analog modulation of laser diodes

We replace the HeNe cavity laser with a 633 nm 100 mW diode laser (HL63163DG, Ushio Inc) driven by a laser diode driver (TLD001, Thorlabs). The LuxX 488 nm laser supports analog modulation but requires rewiring in the connector.

To provide access to the analog modulation of the LuxX laser, we made a

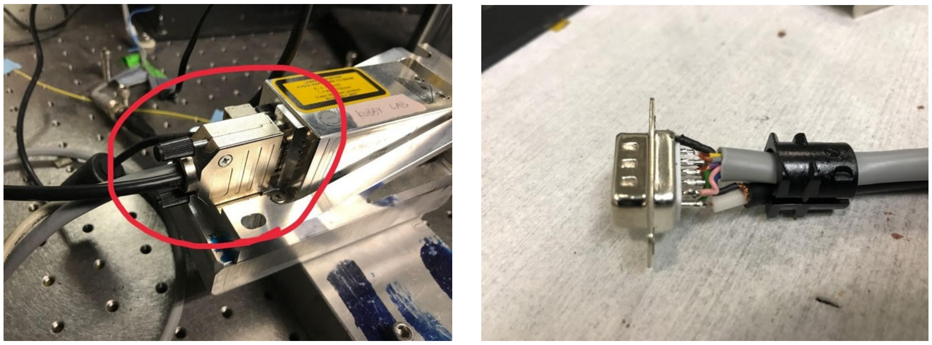


Figure 5.8: Control port connector for LuxX DL.

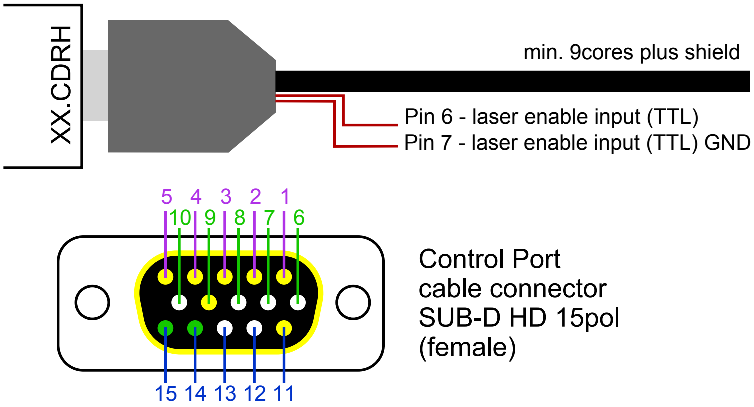


Figure 5.9: Pinout of the connector for LuxX DL.

SUB-D HD 15pol female	Function
1	+5V out Interlock Loop (fused with polyswitch 1A fuse)
2	Interlock Loop Return
3	Key switch (Laser ON/OFF)
4	+5V out (fused with polyswitch 1A fuse)
5	GND
6	Laser enable input (TTL)
7	Digital (TTL) GND for use with Laser enable input
8	RxD / RS232
9	AUX (TTL) (used for CDRH)
10	TxD / RS232
11	Emission LED out (20mA)
12	Error LED out (20mA)
13	Interlock LED out (20mA)
14	Analogue Modulation In (0..5V high impedance)
15	Analogue Modulation GND
Shield	PE / Screen

Table 5.1: Control port pin descriptions.

custom connector cable with two additional wires shown as green rows in Table 5.1. The yellow rows are the wirings from the original cable provided with the laser. This pair of analog modulation wires are connected to a BNC cable to plug into a function generator to provide modulation waveforms. Similarly, we connect the TLC001 driver box to another channel on the function generator via a BNC cable to have modulation of both lasers done via a single 2-channel function generator (33522A, Agilent).

Fig. 5.10 shows the modified version of the TWDM setup with two laser diodes replacing the previous lasers and a function generator to generate analog modulation signals. We use an offset of 95% for the analog modulation pulses for both channels so that we can have lasers on all the time. We found that 10% peak-to-valley modulation

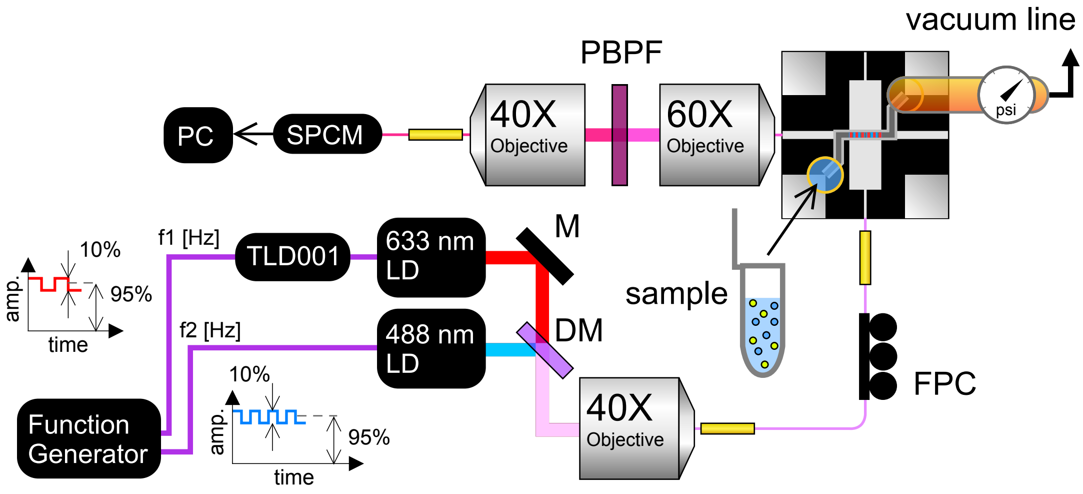


Figure 5.10: TWDM setup with analog modulation.

is enough to form bright bands in the analog regime.

Unlike the chopper blades, with the analog modulation setup, there will be no ‘off’ state for the lasers. As shown in Fig. 5.11, the function generator generates square pulses with specific frequencies for each channel (f_1, f_2) and a DC offset. This DC offset ensures exciting any possible flowing fluorescent bead at any time during the experiment, which tackles the problem of missing events in the chopper setup. Through experiment, we realized that a 10% peak-to-peak modulation is enough to create frequency components in higher concentrations.

Fig. 5.11 illustrates three modes of operation. We consider an intermediate regime where the number of fluorescent beads inside the excitation volume is more than one but less than the analog regime. In this mode, the APD signal can have big fluctuations from zero to a summation of multiple events. The APD signals from three

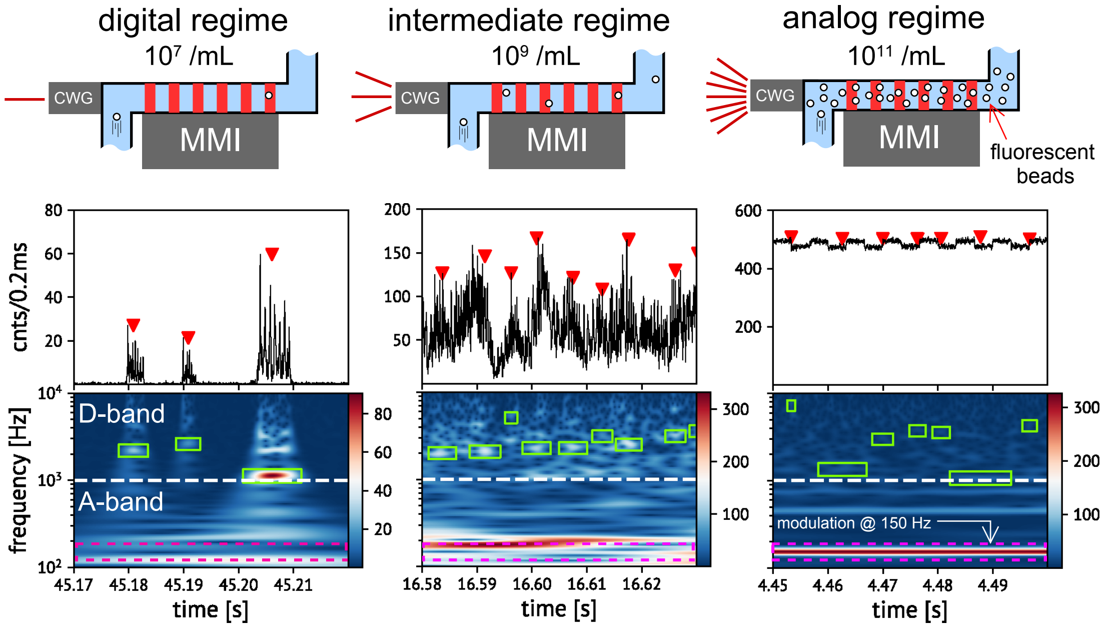


Figure 5.11: Operation regimes in analog modulation TWDM setup.

different concentrations representing three possible modes of operation are shown in Fig. 5.11 with the corresponding spectrogram. The sample used is 200 nm crimson fluorescent beads that are excited by a 633 nm modulated laser diode. As seen, in the digital regime, individual events are detected without any 'off' state and no modulation component in the analog band. The analog regime, on the other hand, features a distinct bright band at the modulation frequency (150 Hz). The events detected in the analog regime are not valid and are random local maxima from the square pulse edges. As expected, the intermediate regime behaves partially similar to the digital and partially similar to the analog regimes. The detected events are not as random as in the analog regime but are not quite clear, and the modulation band is not constant and uniform, therefore a hybrid estimation should be considered in the intermediate regime.

5.3.1 Concentration estimation: analog regime

In order to derive an equation for concentration approximation in the analog regime, we start with writing the CWT transform formula for the recorded APD signal.

$$C(t, \Delta t) = \int_{-\infty}^{+\infty} S(t') \frac{1}{\sqrt{\Delta t}} \psi^* \left(\frac{t - t'}{\Delta t} \right) dt' \tag{5.3}$$

we can break down the APD signal into a product of the input laser modulation ($M(t)$) with the summation of the background and fluorescence emission signals ($B(t), E(t)$). Then the Eq. 5.3 can be rewritten as

$$C(t, \Delta t) = \int_{-\infty}^{+\infty} \left[M(t') \times \left(B(t') + E(t') \right) \right] \frac{1}{\sqrt{\Delta t}} \psi^* \left(\frac{t - t'}{\Delta t} \right) dt' \tag{5.4}$$

we can assume that B and E are constant in the analog regime so they will come out of the integral. Since the emission signal E depends linearly on the target concentration c, we can relate the CWT coefficient at modulation frequency f, C_f, to the sample concentration by

$$\begin{aligned} C(t, \Delta t) &= B \int_{-\infty}^{+\infty} M(t') \frac{1}{\sqrt{\Delta t}} \psi^* \left(\frac{t - t'}{\Delta t} \right) dt' + E \int_{-\infty}^{+\infty} M(t') \frac{1}{\sqrt{\Delta t}} \psi^* \left(\frac{t - t'}{\Delta t} \right) dt' \\ &= b + cE_0 \int_{-\infty}^{+\infty} M(t') \frac{1}{\sqrt{\Delta t}} \psi^* \left(\frac{t - t'}{\Delta t} \right) dt' \\ &= b + ca \\ C(t, f) &= C_f = ac + b \end{aligned} \tag{5.5}$$

$$c = \frac{C_f - b}{a} \tag{5.6}$$

a and b are slopes and background offset parameters, respectively. They are tuned from the calibration experiment by running a series of known concentrations into the chip and mapping C_f values to the expected concentration values. We found $a = 5 \times 10^{-8}$ and $b = 0.1$ for the fluorescence signals used in this study and kept them fixed throughout all analyses.

If we use both digital and analog concentration values calculated using equations 5.1 and 5.6, we end up creating a concentration curve shown in Fig. 5.12. There are two independent curves plotted for digital and analog data and the way that we relate these values to the final estimated concentration is based on the $C_{D,max}$ and calibration experiment. The $C_{D,max}$ determines up to which concentration the digital calculation is valid (plotted as a vertical dashed line in the bottom plot). The horizontal dashed line in the bottom plot represents the threshold where any CWT value above that will be valid and the analog calculation will be considered as the final estimated concentration (highlighted by yellow). As seen previously in Fig. 5.11, there are digital events and C_f values for all regimes and the way that we estimate the concentration. The calibration curve shows how digital concentration data points get saturated after $C_{D,max}$ and how analog values can be used to extend the working range. The analog values in the lower concentration range (digital regime), are mostly from the background signal and remain constant up to $C_{D,max}$ level. The intermediate regime is defined once from the calibration experiment by finding the overlapping region of digital and analog

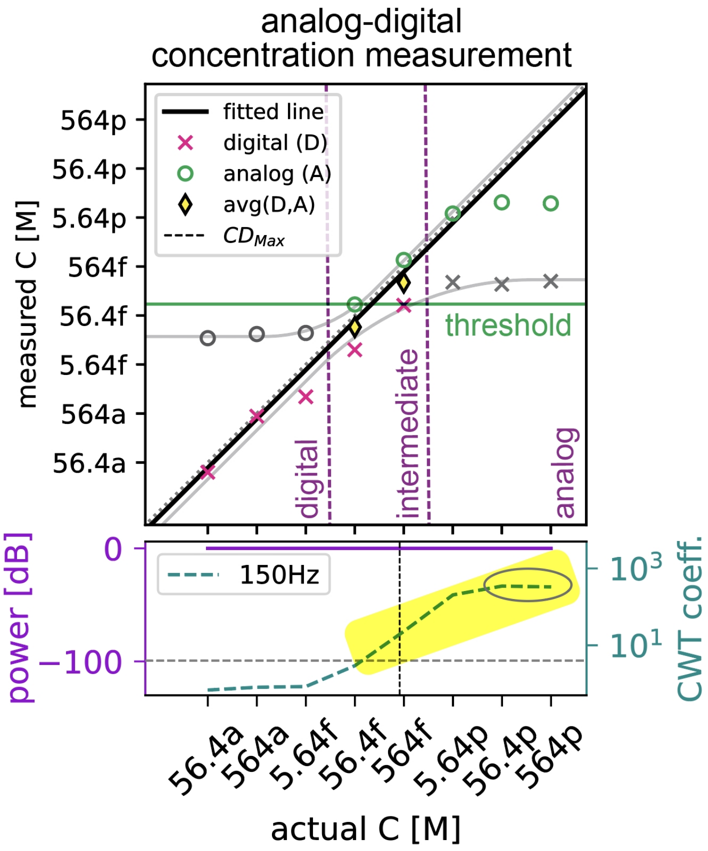


Figure 5.12: High dynamic range concentration estimation via analog TDM modulation.

curves. There is saturation seen in the upper bound of the analog curve which is related to the APD saturation. The detector used in the study is a single-photon-counting module (SPCM-AQRH, Excelitas Technologies) with typical saturation happening at around 5 Mcps.

We repeated the same experiment with a different sample, this time 100 nm yellow-green fluorescent beads. Similar digital limit and analog extension are observed with better performance in the upper bound. This is due to the fact that 100 nm yellow-green beads are less brighter than 200 nm crimson beads and the analog signal,

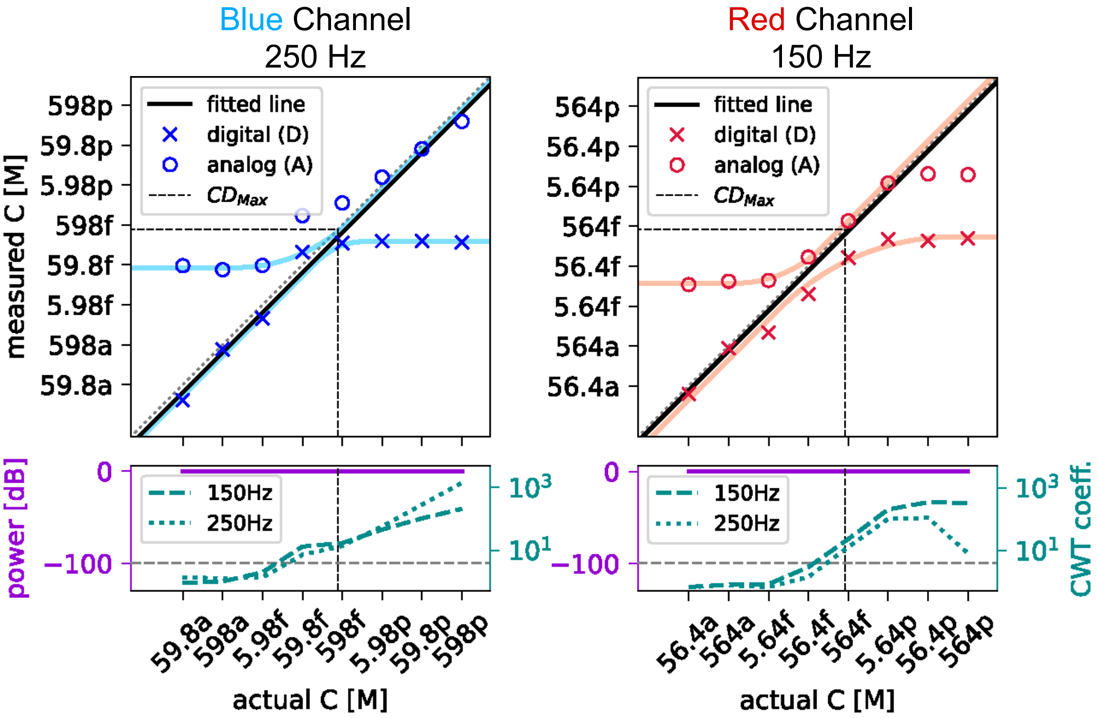


Figure 5.13: High dynamic range curves for singleplex blue and red channels.

even at very high concentrations, does not saturate the SPCM (see Fig. 5.13).

5.4 Multiplexed detection

We investigate the multiplexing capability by introducing a mixture of two target particles, 200 nm crimson, and 100 nm yellow-green fluorescent beads. Fig. 5.14 depicts a multiplexing detection test on a digital regime with both targets sitting in lower concentrations (10^6 beads/mL). As seen, individual events are easily picked and identified with a combination of PCWA event detector and DNN classifier illustrated in Fig. 5.15 (more detail discussed in Chapter 4). In higher concentrations (10^{10}

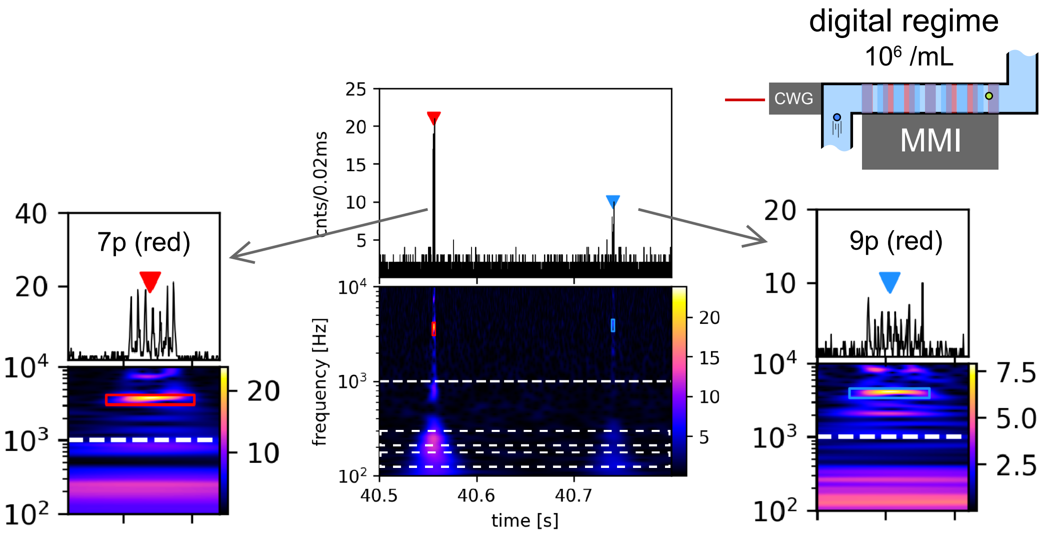


Figure 5.14: Multiplexed detection in digital regime with analog modulation.

beads/mL) analog bands form at modulation frequencies 150 Hz and 250 Hz for red and blue channels respectively.

With the discussed configuration we face some limitations. One was discussed in the crimson detection at higher concentrations and the SPCM saturation. The other one is when the mixed sample includes some highly concentrated targets alongside some low concentrated targets making it hard to see digital events for the lower concentration. In the next section, we introduce an adaptive mechanism to make the system robust and linear in a wider dynamic range.

5.5 Adaptive modulation

In this section, we explore an adaptive design for multiplexed TWDM multiplexed detection. The design utilizes a real-time closed-loop feedback system to analyze

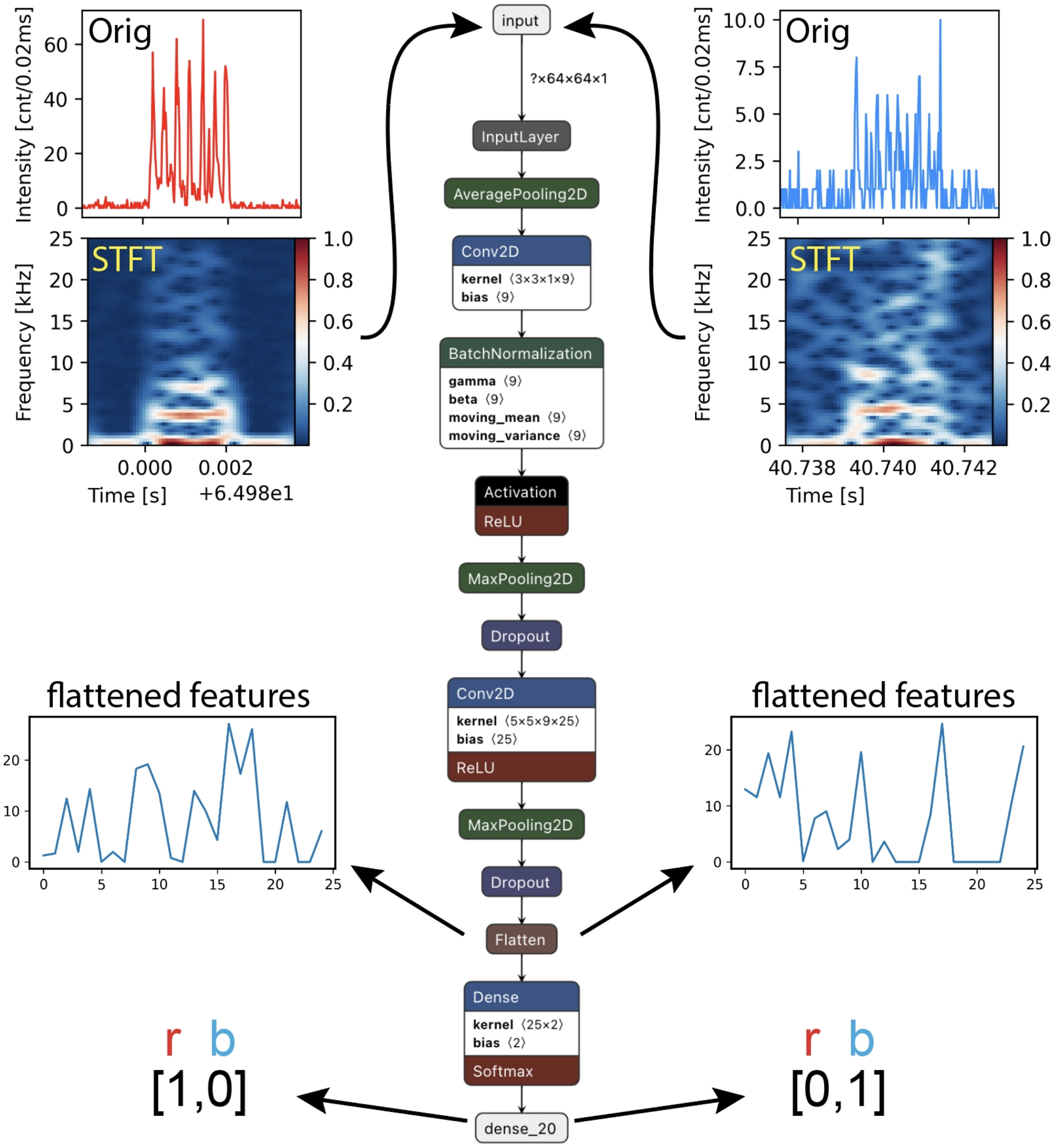


Figure 5.15: DNN classifier for multiplexed detection in digital.

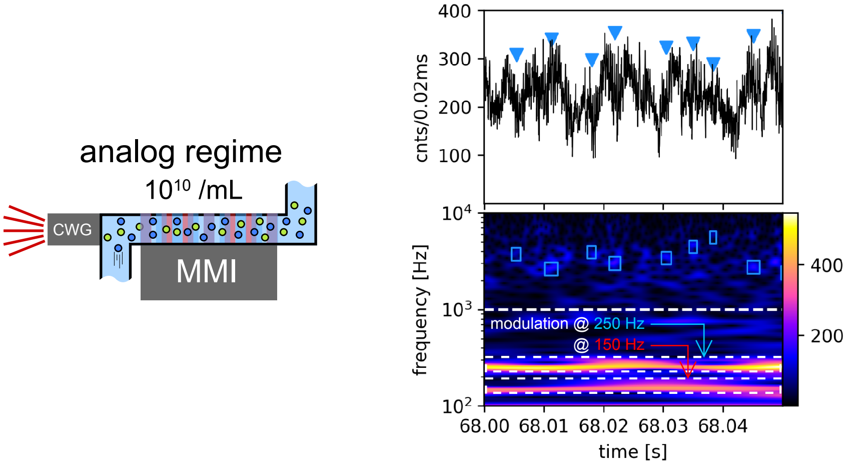


Figure 5.16: Multiplexed detection in analog regime with analog modulation.

the fluorescence signal and adjust the input laser power accordingly. Unlike the previous setup where a time trace was recorded during the experiment and the analysis was carried out later, the adaptive configuration estimates concentration based on the adjusted powers of the laser, hence the real-time feedback is a key element of the system. The adaptive TWDM setup is shown in Fig. 5.17 with a minor structure change of controlling function generator via PC.

The photodetector used in the single-molecule detection setup comes with a high-resolution time-correlated single photon counting module (TCSPC). TimeHarp 260nano (PicoQuant) records timestamps of any detected photons from a single photon detector into a *.tpu file in T2 mode with 250 ps time resolution [91]. We developed a Python GUI program (see Fig. 5.18) to acquire, analyze, and provide feedback signals in real-time for adaptive TWDM purpose. There are visualization, experiment control, and result plot elements implemented into the GUI by the PySide6 library built on

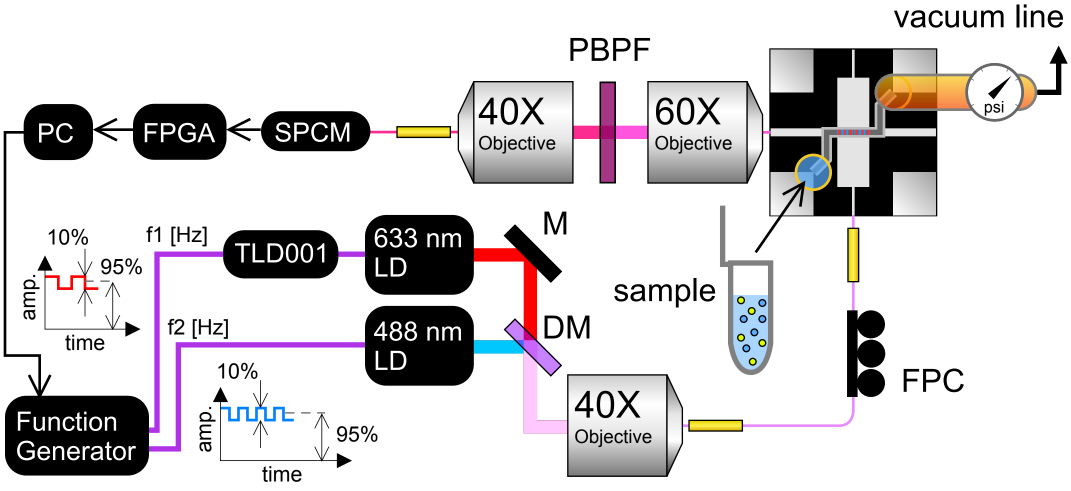


Figure 5.17: Adaptive TWDM setup.

top of Qt6. Even though the program works fine with low-concentration samples, it freezes in high concentrations samples due to buffer overflow from timeharp260 DAQ library. This is the bottleneck of binning all incoming photon event timestamps for further data analysis. To solve this problem, we implement a binning unit using a field programmable gate arrays (FPGA) development board.

5.5.1 Real-time photon-counting with FPGA

The idea of using an FPGA instead of timeharp260 card is to offload the binning of photon events from the main CPU into an FPGA. The binned time trace is then transferred to the PC via ethernet (socket protocol) and the Python GUI will receive and analyze the incoming packets in real-time. Another benefit of using FPGA over other solutions like GPU is that this solution is not limited to a PC as the host and can be connected to any other portable host device (i.e. a Raspberry or Google Coral dev board) as long as they have an ethernet interface. Fig. 5.18 shows the schematic of the developed binning algorithm for the Arty-A7 FPGA board [92]. The program shown in Fig. 5.18 is part of a bigger logic, but the key parts are highlighted and explained with red circles.

In Fig. 5.19 we see the experimental setup with fundamental parts highlighted.

ARROW devices are prone to analyte channel clogging and this problem is more likely to happen in higher concentrations, requiring unclogging the chip with solvent. We replace the conventional wax-based reservoir bonding with the thermoplastic bonding of silicone tubings to prevent wax contamination and additional clogging caused by leftover waxes inside the channel (Fig. 5.20). The silicone tubing pieces also are more flexible, hence a better sealing to the 3D-printed chip adapter without the risk of breaking the analyte channel or optical waveguides.

5.6 A/D mixed concentrations and adaptive TWDM

Real-time adaptive mechanism monitors spectral information at modulation frequencies (channels) through short-time Fourier transform (STFT) and maintains their absolute value around a set point. Fig. 5.21 explains the cycle of power adjustment inside the closed-loop system through phases **a**, **b**, and **c**. In this example, at time t_0 feedback system is engaged and a high value of frequency component at f_1 is immediately observed (C_2). This introduces a negative control signal by a proportional-

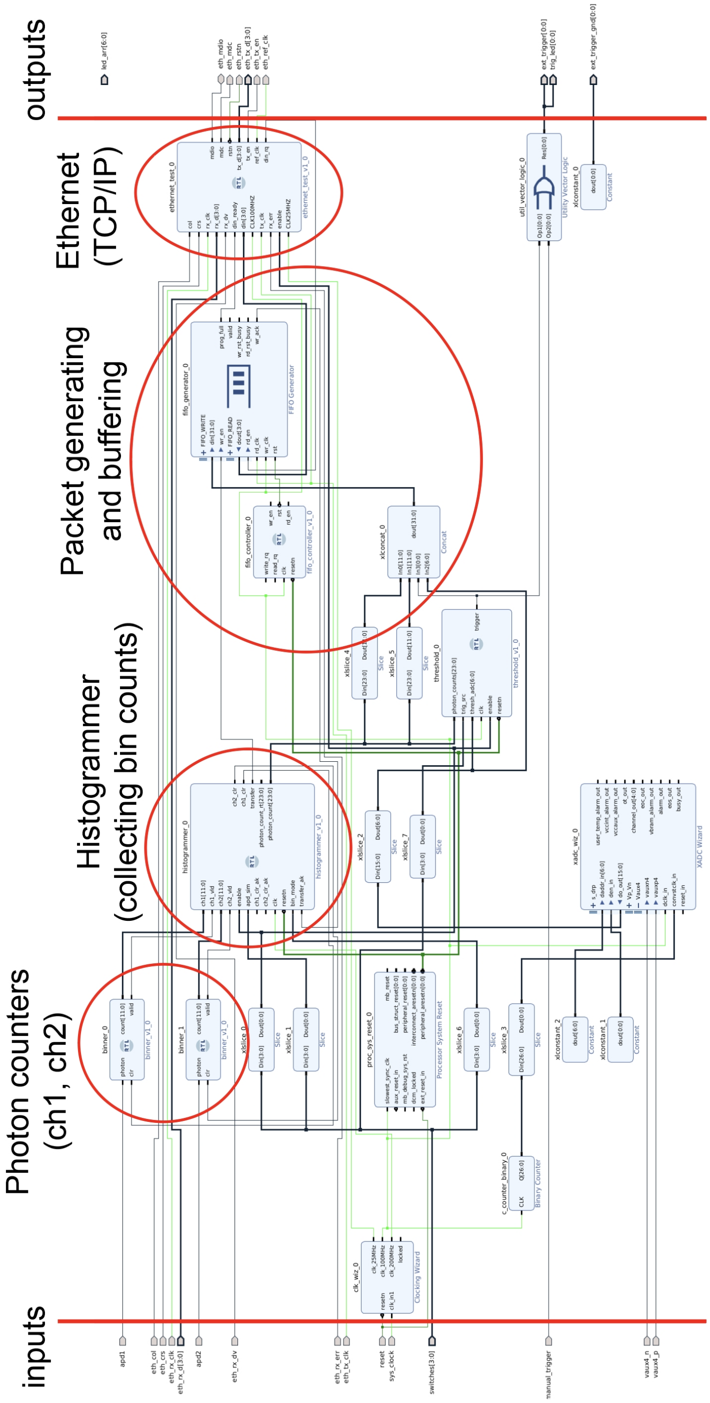


Figure 5.18: Real-time photon counting via FPGA.

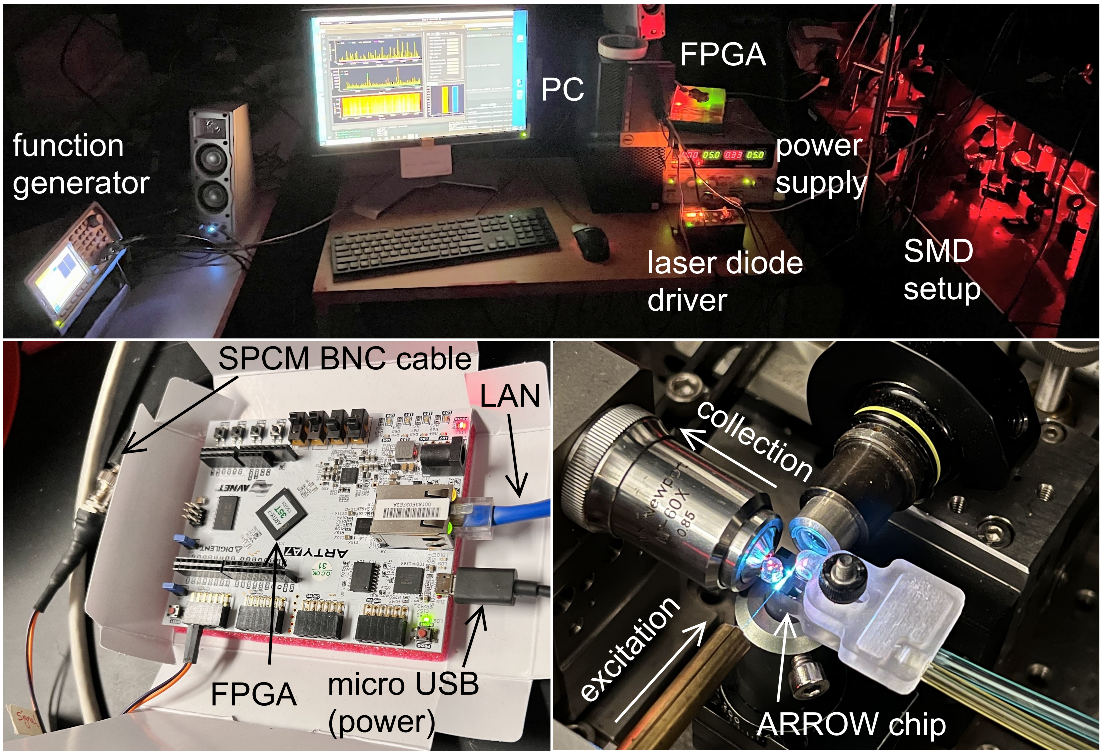


Figure 5.19: Experimental setup parts.

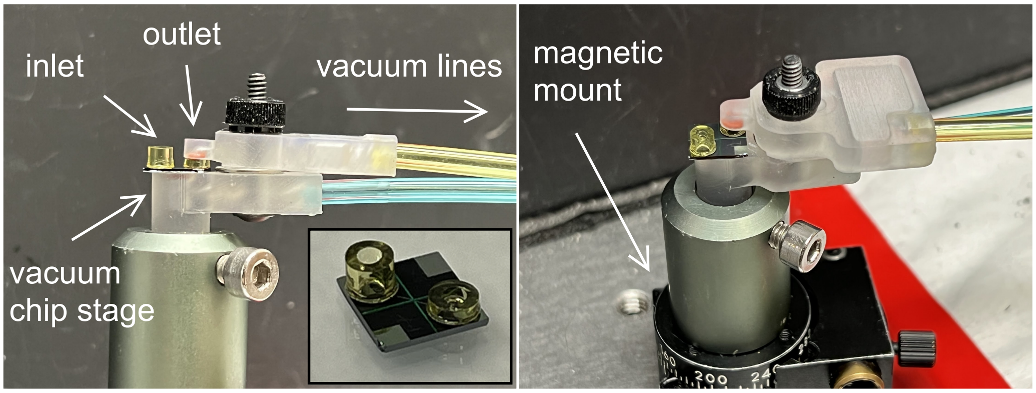


Figure 5.20: 3D printed chip mount.

integral-derivative (PID) system shown in Fig. 5.21b and reduces the laser power at the corresponding channel (Ch_2).

When operating in adaptive mode, Eq. 5.6 for determining concentration in the analog regime requires modification. Specifically, a PID-normalized power adjustment factor, denoted as p_i, is introduced for each channel i. During the experiment, the closed-loop feedback control unit continuously adjusts the modulation pulses of the function generator to ensure the frequency component of each channel remains close to a predetermined setpoint. As a result, the value of p_i varies over time.

$$C = \frac{C_i - ap_i}{bp_i} \tag{5.7}$$

The APD signal, and thus C_i, is proportional to the input laser power p_i, which means that the same derived values for a and b parameters from the analog regime can be used in the adaptive mode. However, if there are changes in the experimental setup or fluorescence properties of the targets, these values may need to be updated.

When the concentration series in Fig. 5.13 was re-evaluated using the adaptive feedback circuit, previous results were reproduced at low and intermediate concentrations. However, at the high concentration end, saturation effects were removed, and linearity was restored over the entire 8-log concentration range. The impact of the adaptive circuit is illustrated in Fig. 5.22 (bottom), which shows a reduction in excitation laser power (purple highlighted by the yellow box) at high powers to maintain the CWT amplitudes safely outside the saturation range (blue curve).

The PID controller is followed by a non-linear system that can respond quickly

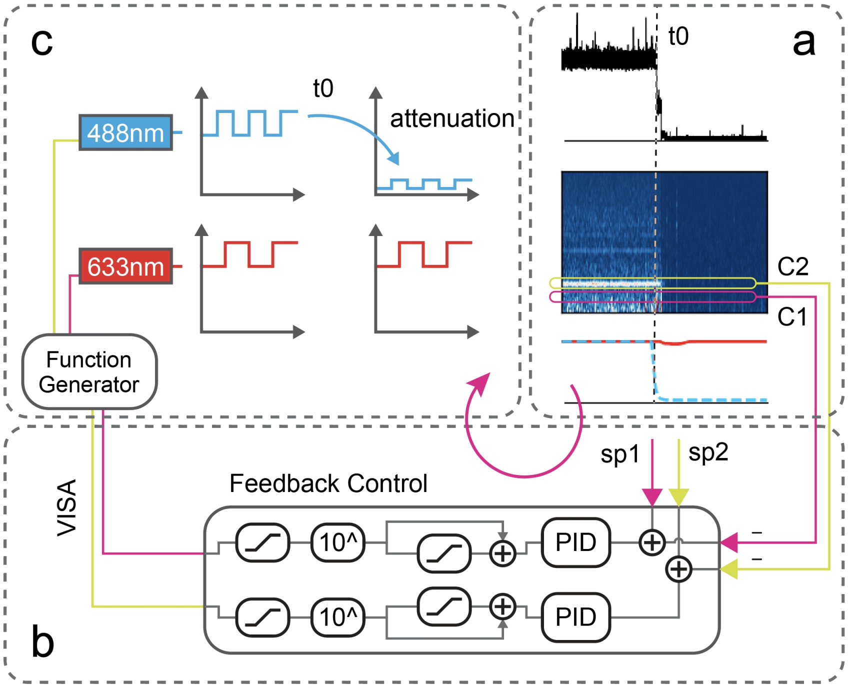


Figure 5.21: Real-time photon counting via FPGA.

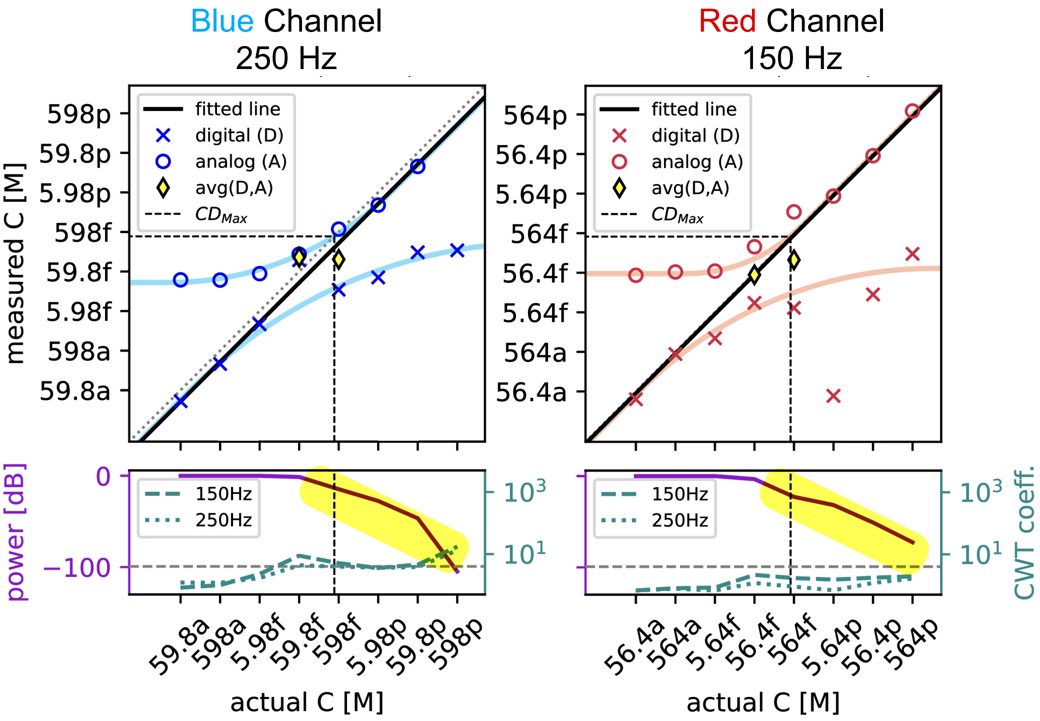


Figure 5.22: Adaptive method to estimate concentrations in high-dynamic range.

to sudden changes in the concentration and maintain the CWT amplitude by dynamically adjusting the power level. It is also capable of recovering back to the digital mode by itself if the concentration decreases.

In an analysis of a challenging mixture of crimson and yellow-green fluorescent nanoparticles at low and high concentrations, respectively, adaptive modulation was used (Fig. 5.23). Initially (t $<$ 9 s), the adaptive feedback circuit was off, and a large fluorescence signal by the high-concentration yellow-green beads did not allow for the reliable detection of any red beads. As soon as the adaptive mode was activated, the blue excitation power was adjusted within a few seconds, and the fluorescence trace displayed individual signals from crimson beads in the digital range while still allowing for determining the yellow-green bead concentration. This is also illustrated by the power attenuation and confidence traces at the bottom panel of Fig. 5.23.

Fig. 5.24 shows how both the yellow-green and crimson bead concentrations were correctly identified, matching their respective singleplex curves, even though their concentration differed by $> 10,000\times$ in the mixture. The estimated concentrations from the mixed solution are shown with start markers and they align nicely with the digital and analog curves from the singleplex data. The digital events in each section of the time trace are classified using the previously trained DNN model.

Overall, the use of adaptive modulation and the closed-loop feedback control unit has allowed for a more accurate and reliable determination of concentration, especially at high concentrations where saturation effects can occur. Additionally, the ability of the system to adapt to changes in concentration quickly and efficiently makes

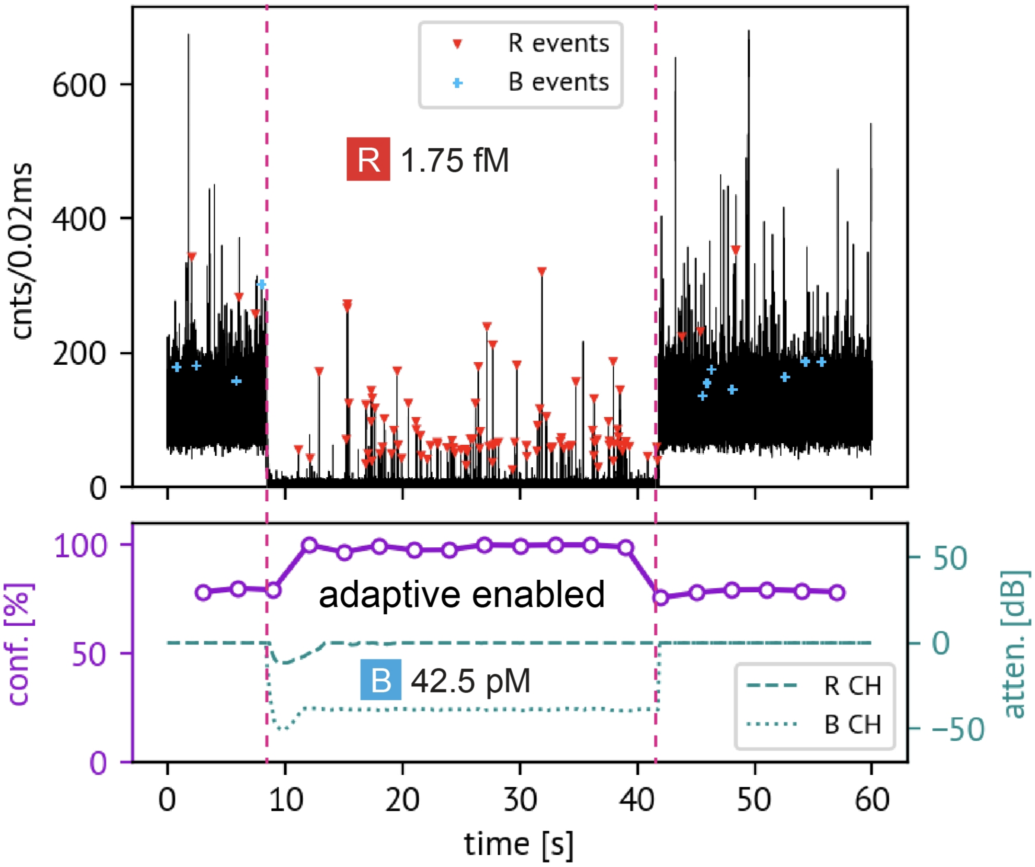


Figure 5.23: A/D mixed sample time trace.

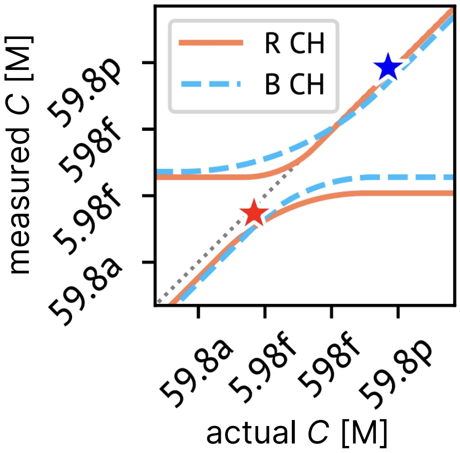


Figure 5.24: A/D concentration estimation.

it a powerful tool in a variety of analytical applications.

Chapter 6

Conclusion

In conclusion, hardware advances as well as incorporating available data analysis and developing new tools to suit the optofluidic needs, result in outstanding performance, especially for biomedical diagnosis and point-of-care applications.

Hydrodynamically focusing the stream of targets inside the ARROW chip was investigated by a trench design and a successful 3D-HDF was reported. Challenges regarding the characterization and analysis of focusing behavior inside the analyte channel, lead to elaborating novel techniques of determining 3D focusing effect based on 2D fluorescence imaging. Also, multiple-channel access within an integrated chip footprint requires a special interfacing mechanism. Advancements in 3D printing technology solved the problem of complex channel routing within small, compact, and custom configurations to drive a 3D-HDF chip.

A fast CWT-based event detection algorithm helped with efficiently detecting more events with higher accuracy not only in fluorescence detection but also applicable to other kinds of signals. The speedup gained in comparison to previous CWT tools, emphasizes the lack of well-developed tools in the field and will demand further advancements to tackle time-frequency analysis in any form of data origin. This tool has been widely used as a daily data analysis gear for optofluidic event detection needs.

Machine learning undoubtedly has surprised science and technology in the past decades by solving previously unsolvable problems. Taking advantage of this growing field in optofluidic biosensing is another big goal achieved in this thesis. The highest accuracy and unmatched performance were demonstrated on an inexpensive, low-power, portable device. Google Coral Dev board was selected for this project as the AI accelerator for inferencing purposes and the results confirm a negligible drop in performance by switching from a desktop PC to this compact device. The benefits of using edge devices in biomedical diagnosis, such as data privacy concerns, lower bandwidth requirement, and on-demand test results and monitoring to the user are discussed.

Finally, an innovative scheme for driving ARROW chips was introduced to increase the dynamic range of detection. The results show a $10^4\times$ increase in the higher concentration range of detection without any sacrifice observed in the lower end. In order to attain this goal, several modifications in the current single-molecule-detection setup were introduced. For example, real-time acquisition and analysis of fluorescence signals from the SPCM module were needed for a closed-loop system and therefore an FPGA program was developed to do this task. The real-time adjustment of input laser modules independently for two channels was demonstrated to solve another challenging limit of a conventional biosensor which is concentration estimation for a mixture of low- and high-concentration samples. Multiplexing achieved by time and wavelength division of the sensory signal (TWDM) was presented with results highlighting successfully estimating the concentration of two target types with the 4-order difference in concentrations.

Optofluidics is a new technology that blends photonics and microfluidics into an integrated platform for bio-detection. Along with all the unique features and benefits borrowed from optics and microfluidics, it also creates new challenges by blending the ancestor's challenges. Ultrasensitive photonic components functioning within an everchanging dynamic fluidic environment demand novel techniques in science and engineering.

Bibliography

[1] Erik S. Hamilton, Vahid Ganjalizadeh, Joel G. Wright, William G. Pitt, Holger Schmidt, and Aaron R. Hawkins. 3D hydrodynamic focusing in microscale channels formed with two photoresist layers. *Microfluidics and Nanofluidics*, 23(11):1–8, 2019.

[2] Erik S. Hamilton, Vahid Ganjalizadeh, Joel G. Wright, Holger Schmidt, and Aaron R. Hawkins. 3D Hydrodynamic Focusing in Microscale Optofluidic Channels Formed with a Single Sacrificial Layer. *Micromachines*, 11(4):349, April 2020.

[3] Jennifer A. Black, Erik Hamilton, Raúl A. Reyes Hueros, Joshua W. Parks, Aaron R. Hawkins, and Holger Schmidt. Enhanced Detection of Single Viruses On-Chip via Hydrodynamic Focusing. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(1):1–6, January 2019.

[4] L.B. Soldano and E.C.M. Pennings. Optical multi-mode interference devices based on self-imaging: Principles and applications. *Journal of Lightwave Technology*, 13(4):615–627, April 1995.

[5] Matthew A. Stott, Vahid Ganjalizadeh, Maclain Olsen, Marcos Orfila, Johnny

McMurray, Holger Schmidt, and Aaron R. Hawkins. High Fidelity MMI Excitation Patterns for Optofluidic Multiplexing. In *CLEO: QELS_Fundamental Science*, page JW2A. 24. Optical Society of America, 2018.

[6] Matthew A. Stott, Vahid Ganjalizadeh, Gopikrishnan Meena, Johnny McMurray, Maclain Olsen, Marcos Orfila, Holger Schmidt, and Aaron R. Hawkins. Buried Rib SiO 2 Multimode Interference Waveguides for Optofluidic Multiplexing. *IEEE Photonics Technology Letters*, 30(16):1487–1490, 2018.

[7] H. Schmidt, Dongliang Yin, J.P. Barber, and A.R. Hawkins. Hollow-core waveguides and 2-D waveguide arrays for integrated optics of gases and liquids. *IEEE Journal of Selected Topics in Quantum Electronics*, 11(2):519–527, March 2005.

[8] Joshua W. Parks and Holger Schmidt. Flexible optofluidic waveguide platform with multi-dimensional reconfigurability. *Scientific Reports*, 6(1):33008, September 2016.

[9] Gopikrishnan G. Meena, Aadhar Jain, Joshua W. Parks, Alexandra Stambaugh, Jean L. Patterson, Aaron R. Hawkins, and Holger Schmidt. Integration of sample preparation and analysis into an optofluidic chip for multi-target disease detection. *Lab on a Chip*, 18(23):3678–3686, 2018.

[10] Jennifer A. Black, Vahid Ganjalizadeh, Joshua W. Parks, and Holger Schmidt. Multi-channel velocity multiplexing of single virus detection on an optofluidic chip. *Optics Letters*, 43(18):4425–4428, September 2018.

[11] D.A. Chang-Yen, R.K. Eich, and B.K. Gale. A monolithic PDMS waveguide system fabricated using soft-lithography techniques. *Journal of Lightwave Technology*, 23(6):2088–2093, June 2005.

[12] Joshua W. Parks. *Hybrid Optofluidic Biosensors.* University of California, Santa Cruz, 2016.

[13] W. P. Risk, H.-C. Kim, R. D. Miller, H. Temkin, and S. Gangopadhyay. Optical waveguides with an aqueous core and a low-index nanoporous cladding. *Optics express*, 12(26):6446–6455, 2004.

[14] Dongliang Yin, Holger Schmidt, John P. Barber, and Aaron R. Hawkins. Integrated ARROW waveguides with hollow cores. *Optics Express*, 12(12):2710–2715, 2004.

[15] Claire Simonnet and Alex Groisman. Two-dimensional hydrodynamic focusing in a simple microfluidic device. *Applied Physics Letters*, 87(11):114104, September 2005.

[16] Jennifer A. Black, Erik Hamilton, Raúl A. Reyes Hueros, Joshua W. Parks, Aaron R. Hawkins, and Holger Schmidt. Enhanced Detection of Single Viruses On-Chip via Hydrodynamic Focusing. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(1):1–6, January 2019.

[17] Michael A. Daniele, Darryl A. Boyd, David R. Mott, and Frances S. Ligler. 3D hydrodynamic focusing microfluidics for emerging sensing technologies. *Biosensors and Bioelectronics*, 67:25–34, May 2015.

[18] Hamed Amini, Elodie Sollier, Mahdokht Masaeli, Yu Xie, Baskar Ganapathysubramanian, Howard A. Stone, and Dino Di Carlo. Engineering fluid flow using sequenced microstructures. *Nature Communications*, 4(1):1826, May 2013.

[19] Jennifer A. Black, Erik Hamilton, Raúl A. Reyes Hueros, Joshua W. Parks, Aaron R. Hawkins, and Holger Schmidt. Enhanced Detection of Single Viruses On-Chip via Hydrodynamic Focusing. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(1):1–6, January 2019.

[20] Myung Gwon Lee, Sungyoung Choi, and Je-Kyun Park. Three-dimensional hydrodynamic focusing with a single sheath flow in a single-layer microfluidic device. *Lab on a Chip*, 9(21):3155–3160, November 2009.

[21] Xiaole Mao, John Robert Waldeisen, and Tony Jun Huang. "Microfluidic drifting"—implementing three-dimensional hydrodynamic focusing with a single-layer planar microfluidic device. *Lab on a Chip*, 7(10):1260–1262, September 2007.

[22] Yu-Jui Chiu, Sung Hwan Cho, Zhe Mei, Victor Lien, Tsung-Feng Wu, and Yu-Hwa Lo. Universally applicable three-dimensional hydrodynamic microfluidic flow focusing. *Lab on a Chip*, 13(9):1803–1809, April 2013.

[23] Erik S. Hamilton, Joel G. Wright, Matthew A. Stott, Jennifer A. Black, Holger Schmidt, and Aaron R. Hawkins. 3D Hydrodynamic Focusing for Optofluidics Using a Stacked Channel Design. In *Conference on Lasers and Electro-Optics (2018), Paper JTh2A.101*, page JTh2A.101. Optica Publishing Group, May 2018.

[24] Genni Testa and Romeo Bernini. Integrated tunable liquid optical fiber. *Lab on a Chip*, 12(19):3670–3672, August 2012.

[25] John Conrad Jaeger and Horatio Scott Carslaw. *Conduction of Heat in Solids*. Clarendon P Oxford, United Kingdom:, 1959.

[26] Peter Kapusta. Absolute diffusion coefficients: Compilation of reference data for FCS calibration. *PicoQuant GmbH*, 1:1–2, 2010.

[27] Earle Stellwagen, Yongjun Lu, and Nancy C. Stellwagen. Unified description of electrophoresis and diffusion for DNA and other polyions. *Biochemistry*, 42(40):11745–11750, 2003.

[28] Vahid Ganjalizadeh and Holger Schmidt. PCWA: A fast custom wavelet analysis technique for single molecule detection and identification. Zenodo, December 2021.

[29] Luc Gervais, Nico de Rooij, and Emmanuel Delamarche. Microfluidic Chips for Point-of-Care Immunodiagnostics. *Advanced Materials*, 23(24):H151–H176, 2011.

[30] Hamed Tavakoli, Samayeh Mohammadi, Xiaochun Li, Guanglei Fu, and XiuJun Li. Microfluidic platforms integrated with nano-sensors for point-of-care bioanalysis. *TrAC Trends in Analytical Chemistry*, 157:116806, December 2022.

[31] Xudong Fan and Ian M. White. Optofluidic microsystems for chemical and biological analysis. *Nature Photonics*, 5(10):591–597, October 2011.

[32] Holger Schmidt and Aaron R. Hawkins. The photonic integration of non-solid media using optofluidics. *Nature Photonics*, 5(10):598–604, October 2011.

[33] H. Cai, J. W. Parks, T. A. Wall, M. A. Stott, A. Stambaugh, K. Alfson, A. Griffiths, R. A. Mathies, R. Carrion, J. L. Patterson, A. R. Hawkins, and H. Schmidt. Optofluidic analysis system for amplification-free, direct detection of Ebola infection. *Scientific Reports*, 5(1):14494, September 2015.

[34] Alexandra M. Stambaugh, Matthew A. Stott, Gopikrishnan G. Meena, Manasi Tamhankar, Ricardo Carrion, Jean L. Patterson, Aaron R. Hawkins, and Holger Schmidt. Optofluidic Amplification-Free Multiplex Detection of Viral Hemorrhagic Fevers. *IEEE Journal of Selected Topics in Quantum Electronics*, 27(4):1–6, July 2021.

[35] H. Cai, J. W. Parks, T. A. Wall, M. A. Stott, A. Stambaugh, K. Alfson, A. Griffiths, R. A. Mathies, R. Carrion, J. L. Patterson, A. R. Hawkins, and H. Schmidt. Optofluidic analysis system for amplification-free, direct detection of Ebola infection. *Scientific Reports*, 5(1):14494, September 2015.

[36] Damla Ozcelik, Joshua W. Parks, Thomas A. Wall, Matthew A. Stott, Hong Cai, Joseph W. Parks, Aaron R. Hawkins, and Holger Schmidt. Optofluidic wavelength division multiplexing for single-virus detection. *Proceedings of the National Academy of Sciences*, 112(42):12933–12937, October 2015.

[37] Damla Ozcelik, Matthew A. Stott, Joshua W. Parks, Jennifer A. Black, Thomas A. Wall, Aaron R. Hawkins, and Holger Schmidt. Signal-to-Noise Enhancement in Optical Detection of Single Viruses With Multispot Excitation. *IEEE Journal of Selected Topics in Quantum Electronics*, 22(4):6–11, July 2016.

[38] Shuo Liu, Thomas A. Wall, Damla Ozcelik, Joshua W. Parks, Aaron R. Hawkins, and Holger Schmidt. Electro-optical detection of single λ-DNA. *Chemical Communications*, 51(11):2084–2087, January 2015.

[39] John M. Gregoire, Darren Dale, and R. Bruce van Dover. A wavelet transform algorithm for peak detection and application to powder x-ray diffraction data. *Review of Scientific Instruments*, 82(1):015105, January 2011.

[40] Oliver Faust, U. Rajendra Acharya, Hojjat Adeli, and Amir Adeli. Wavelet-based EEG processing for computer-aided seizure detection and epilepsy diagnosis. *Seizure*, 26:56–64, March 2015.

[41] Ralph Meier, Heike Dittrich, Andreas Schulze-Bonhage, and Ad Aertsen. Detecting Epileptic Seizures in Long-term Human EEG: A New Approach to Automatic Online and Real-Time Detection and Classification of Polymorphic Seizure Patterns. *Journal of Clinical Neurophysiology*, 25(3):119–131, June 2008.

[42] John E. Ball and Alan Tolley. Low SNR radar signal detection using the continuous wavelet transform (CWT) and a Morlet wavelet. In *2008 IEEE Radar Conference*, pages 1–6, May 2008.

[43] Kaz Adamowski, Andreas Prokoph, and Jan Adamowski. Development of a new method of wavelet aided trend detection and estimation. *Hydrological Processes*, 23(18):2686–2696, 2009.

[44] A.S. Lewis and G. Knowles. Image compression using the 2-D wavelet transform. *IEEE Transactions on Image Processing*, 1(2):244–250, April 1992.

[45] Pan Du, Warren A. Kibbe, and Simon M. Lin. Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching. *Bioinformatics*, 22(17):2059–2065, September 2006.

[46] Stéphane Mallat. *A Wavelet Tour of Signal Processing.* Elsevier, 1999.

[47] PicoQuant/PicoQuant-Time-Tagged-File-Format-Demos: Demo Code for PicoQuants Time Tagged File Formats. https://github.com/PicoQuant/PicoQuant-Time-Tagged-File-Format-Demos.

[48] Vahid Ganjalizadeh, Gopikrishnan G. Meena, Thomas A. Wall, Matthew A. Stott, Aaron R. Hawkins, and Holger Schmidt. Fast custom wavelet analysis technique for single molecule detection and identification. *Nature Communications*, 13(1):1–9, 2022.

[49] Zhi-Min Zhang, Xia Tong, Ying Peng, Pan Ma, Ming-Jin Zhang, Hong-Mei Lu, Xiao-Qing Chen, and Yi-Zeng Liang. Multiscale peak detection in wavelet space. *Analyst*, 140(23):7955–7964, 2015.

[50] David Deamer, Mark Akeson, and Daniel Branton. Three decades of nanopore sequencing. *Nature Biotechnology*, 34(5):518–524, May 2016.

[51] Bala Murali Venkatesan and Rashid Bashir. Nanopore sensors for nucleic acid analysis. *Nature Nanotechnology*, 6(10):615–624, October 2011.

[52] James Clarke, Hai-Chen Wu, Lakmal Jayasinghe, Alpesh Patel, Stuart Reid, and Hagan Bayley. Continuous base identification for single-molecule nanopore DNA sequencing. *Nature Nanotechnology*, 4(4):265–270, April 2009.

[53] M. Rahman, M. Harrington, M. A. Stott, Y. Li, M. J. N. Sampad, T. D. Yuzvinsky, A. R. Hawkins, and H. Schmidt. Optical trapping assisted detection rate enhancement of single molecules on a nanopore optofluidic chip. *Optica*, 6(9):1130–1131, September 2019.

[54] M. Rahman, M. A. Stott, M. Harrington, Y. Li, M. J. N. Sampad, L. Lancaster, T. D. Yuzvinsky, H. F. Noller, A. R. Hawkins, and H. Schmidt. On demand delivery and analysis of single molecules on a programmable nanopore-optofluidic device. *Nature Communications*, 10(1):3712, August 2019.

[55] Jeffrey S. Morris, Kevin R. Coombes, John Koomen, Keith A. Baggerly, and Ryuji Kobayashi. Feature extraction and quantification for mass spectrometry in biomedical applications using the mean spectrum. *Bioinformatics (Oxford, England)*, 21(9):1764–1775, May 2005.

[56] Peter Kiesel, Michael Bassler, Markus Beck, and Noble Johnson. Spatially modulated fluorescence emission from moving particles. *Applied Physics Letters*, 94(4):041107, January 2009.

[57] Damla Ozcelik, Aadhar Jain, Alexandra Stambaugh, Matthew A. Stott, Joshua W. Parks, Aaron Hawkins, and Holger Schmidt. Scalable Spatial-Spectral Multiplexing

of Single-Virus Detection Using Multimode Interference Waveguides. *Scientific Reports*, 7(1):12199, September 2017.

[58] Joerg Martini, Michael I. Recht, Malte Huck, Marshall W. Bern, Noble M. Johnson, and Peter Kiesel. Time encoded multicolor fluorescence detection in a microfluidic flow cytometer. *Lab on a Chip*, 12(23):5057–5062, 2012.

[59] Pengfei Xie, Xinnan Cao, Zhongtian Lin, Niloy Talukder, Sam Emaminejad, and Mehdi Javanmard. Processing gain and noise in multi-electrode impedance cytometers: Comprehensive electrical design methodology and characterization. *Sensors and Actuators B: Chemical*, 241:672–680, March 2017.

[60] V. Lien, Kai Zhao, Y. Berdichevsky, and Yu-Hwa Lo. High-sensitivity cytometric detection using fluidic-photonic integrated circuits with array waveguides. *IEEE Journal of Selected Topics in Quantum Electronics*, 11(4):827–834, July 2005.

[61] Gopikrishnan Gopalakrishnan Meena. *Multiplexed Biosensors for Pathogen Detection*. PhD thesis, University of California, Santa Cruz, United States – California, 2021.

[62] SPCM-AQRH, Single-Photon Counting Module, Silicon Avalanche Photodiode — Excelitas. https://www.excelitas.com/product/spcm-aqrh.

[63] Sangbae Kim and Byoung S. Ham. Revisiting self-interference in Young's double-slit experiments. *Scientific Reports*, 13(1):977, January 2023.

[64] Victor Lien, Kai Zhao, and Yu-Hwa Lo. Fluidic photonic integrated circuit for in-line detection. *Applied Physics Letters*, 87(19):194106, November 2005.

[65] Ziliang Cai, Weiping Qiu, Guocheng Shao, and Wanjun Wang. A new fabrication method for all-PDMS waveguides. *Sensors and Actuators A: Physical*, 204:44–47, December 2013.

[66] Hassan El-Kashef. The necessary requirements imposed on polar dielectric laser dye solvents. *Physica B: Condensed Matter*, 279(4):295–301, May 2000.

[67] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, December 1943.

[68] Leslie G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[69] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, April 1980.

[70] Xiaojun Xiang, Qiong Li, Shahnawaz Khan, and Osamah Ibrahim Khalaf. Urban water resource management for sustainable environment planning using artificial intelligence techniques. *Environmental Impact Assessment Review*, 86:106515, 2021.

[71] Tankiso Moloi and Tshilidzi Marwala. *Artificial Intelligence in Economics and Finance Theories.* Springer, 2020.

[72] Jim Sterne. *Artificial Intelligence for Marketing: Practical Applications.* John Wiley & Sons, 2017.

[73] Kit-Kay Mak and Mallikarjuna Rao Pichika. Artificial intelligence in drug development: Present status and future prospects. *Drug Discovery Today*, 24(3):773–780, March 2019.

[74] M. O. Riedl and R. M. Young. Narrative Planning: Balancing Plot and Character. *Journal of Artificial Intelligence Research*, 39:217–268, September 2010.

[75] Tanha Talaviya, Dhara Shah, Nivedita Patel, Hiteshri Yagnik, and Manan Shah. Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides. *Artificial Intelligence in Agriculture*, 4:58–73, January 2020.

[76] Peter Ström, Kimmo Kartasalo, Henrik Olsson, Leslie Solorzano, Brett Delahunt, Daniel M Berney, David G Bostwick, Andrew J Evans, David J Grignon, Peter A Humphrey, Kenneth A Iczkowski, James G Kench, Glen Kristiansen, Theodorus H van der Kwast, Katia R M Leite, Jesse K McKenney, Jon Oxley, Chin-Chen Pan, Hemamali Samaratunga, John R Srigley, Hiroyuki Takahashi, Toyonori Tsuzuki, Murali Varma, Ming Zhou, Johan Lindberg, Cecilia Lindskog, Pekka Ruusuvuori, Carolina Wählby, Henrik Grönberg, Mattias Rantalainen, Lars Egevad, and Martin Eklund. Artificial intelligence for diagnosis and grading of prostate cancer in biopsies: A population-based, diagnostic study. *The Lancet Oncology*, 21(2):222–232, February 2020.

[77] M. Tanveer, B. Richhariya, R. U. Khan, A. H. Rashid, P. Khanna, M. Prasad, and C. T. Lin. Machine Learning Techniques for the Diagnosis of Alzheimer's Disease: A Review. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 16(1s):30:1–30:35, April 2020.

[78] Feiyun Cui, Yun Yue, Yi Zhang, Ziming Zhang, and H. Susan Zhou. Advancing Biosensors with Machine Learning. *ACS Sensors*, 5(11):3346–3364, November 2020.

[79] Marly F. J. A. van der Lubbe, Akshayaa Vaidyanathan, Marjolein de Wit, Elske L. van den Burg, Alida A. Postma, Tjasse D. Bruintjes, Monique A. L. Bilderbeek-Beckers, Patrick F. M. Dammeijer, Stephanie Vanden Bossche, Vincent Van Rompaey, Philippe Lambin, Marc van Hoof, and Raymond van de Berg. A non-invasive, automated diagnosis of Menière's disease using radiomics and machine learning on conventional magnetic resonance imaging: A multicentric, case-controlled feasibility study. *La radiologia medica*, 127(1):72–82, January 2022.

[80] Jindong Sun, Yanjun Peng, Yanfei Guo, and Dapeng Li. Segmentation of the multimodal brain tumor image used the multi-pathway architecture method based on 3D FCN. *Neurocomputing*, 423:34–45, January 2021.

[81] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.* "O'Reilly Media, Inc.", September 2019.

[82] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017.

[83] Stefan Wager, Sida Wang, and Percy S Liang. Dropout Training as Adaptive Regularization. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[84] Keyan Cao, Yefan Liu, Gongjie Meng, and Qimeng Sun. An Overview on Edge Computing Research. *IEEE Access*, 8:85714–85728, 2020.

[85] Technology. https://coral.ai/technology/.

[86] TensorFlow models on the Edge TPU. https://coral.ai/docs/edgetpu/models-intro/#supported-operations.

[87] G. G. Meena, T. A. Wall, M. A. Stott, O. Brown, R. Robison, A. R. Hawkins, and H. Schmidt. 7X multiplexed, optofluidic detection of nucleic acids for antibiotic-resistance bacterial screening. *Optics Express*, 28(22):33019–33027, October 2020.

[88] Markus Ringnér. What is principal component analysis? *Nature Biotechnology*, 26(3):303–304, March 2008.

[89] Hervé Abdi and Lynne J. Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[90] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008.

[91] Michael Wahl and Sandra Orthaus-Müller. Time tagged time-resolved fluorescence data collection in life sciences. *Technical Note. PicoQuant GmbH, Germany*, 2:1–10, 2014.

[92] Arty A7 Reference Manual - Digilent Reference. https://digilent.com/reference/programmable-logic/arty-a7/reference-manual.

Appendix A

Simulation model for optofluidic particle detection

We developed a simplified model of an optofluidic biosensor to simulate detected event distribution in terms of important metrics such as velocity (Δt in PCWA analysis) and fluorescence signal intensity. The model takes predicted profiles for velocity, excitation and collection into account to generate a dataset of events.

Fig. A.1 shows the scanning electron microscopy (SEM) images of an ARROW chip with important optical waveguides highlighted in Fig. A.1a and a cross-section of the excitation waveguide taken at the chip facet in Fig. A.1b. In Fig. A.2, the chip was milled right at the SM excitation waveguide and analyte channel intersection using focused ion beam (FIB) milling technique. There are possible offsets between waveguides caused by fabrication errors and are shown by guidelines. Fig. A.2a shows the horizontal offset seen between the analyte channel (HC waveguide) and the collection waveguide.

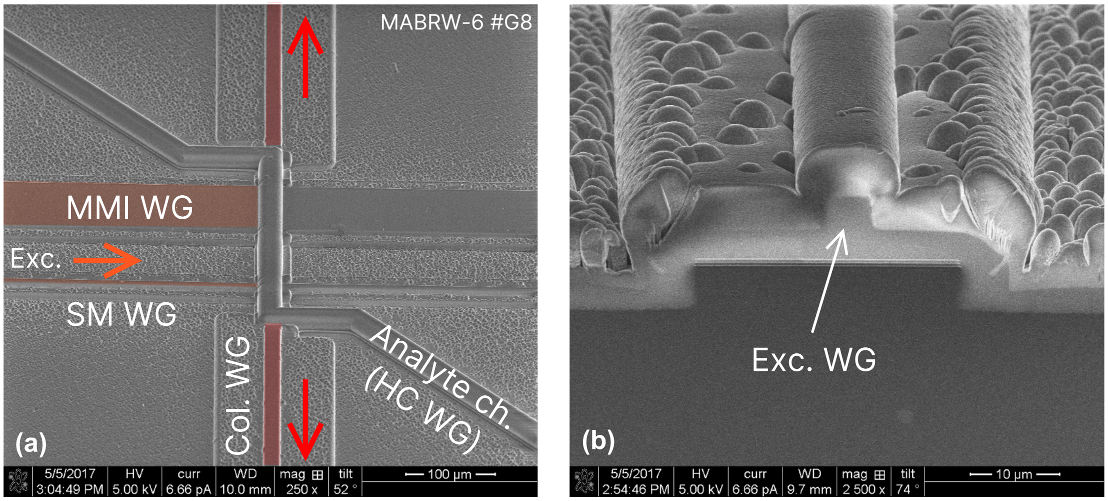


Figure A.1: SEM images of ARROW optofluidic biosensor with key optical and fluidic components.

The vertical offset between the excitation waveguide and the HC waveguide is annotated in Fig. A.2b.

In order to model the conditions each flowing particle can experience, we break down the physics of the system into independent profiles illustrated in Fig. A.3. First, we assume that the particles flowing inside the analyte channel have a parabolic velocity distribution, $u(x, y)$ described as

$$u(x, y) = G \left[1 - \left(\frac{x}{w} \right)^2 \right] \left[1 - \left(\frac{y}{h} \right)^2 \right] \tag{A.1}$$

w and h are the width and height of the channel respectively and G is the maximum velocity at the center of the channel ($x = 0, y = 0$, see Fig. A.3 top-left). Next, we consider a Gaussian approximation of the optical mode profiles for excitation and collection waveguides. For the collection waveguide a 2D version of the Gaussian

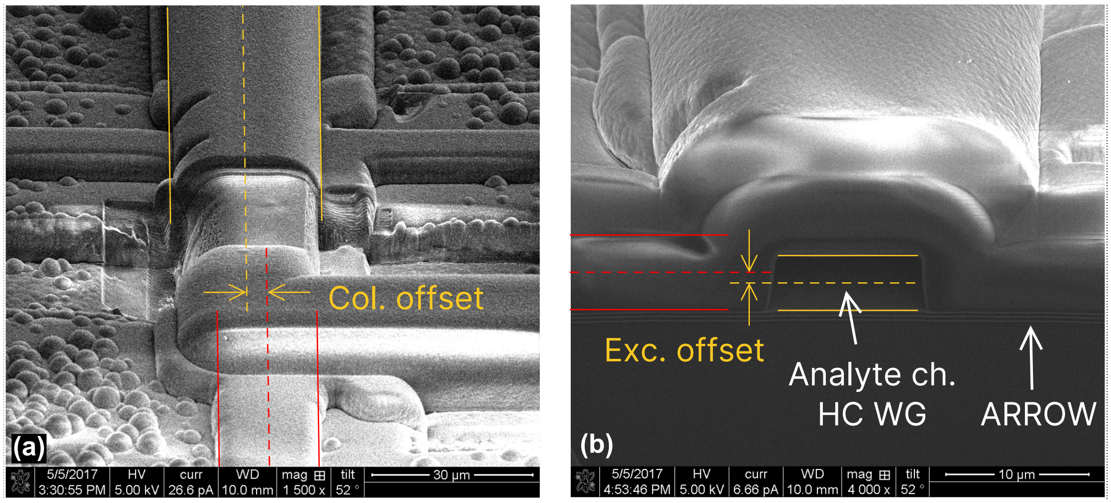


Figure A.2: SEM images of an ARROW chip with possible waveguide offsets.

distribution is applied and for the excitation waveguide, since the information along the z axis is already included in the velocity information (from the velocity profile), a 1D Gaussian distribution is been used.

$$\begin{gathered} Exc(y) = e^{-\left[(y-\mu)^2/2\sigma^2\right]}, \\ Col(x,y) = e^{-\left[(x-\mu_x)^2/2\sigma_x^2+(y-\mu_y)^2/2\sigma_y^2\right]} \end{gathered} \tag{A.2}$$

The intensity of the fluorescent events in an ARROW device depends on how strong the excitation volume excites particles, and how well the fluorescence emission is collected through the collection waveguide. In mathematics, this means a multiplication operation, and we call the result "sensitivity profile". Fig. A.4 visualizes how 1D excitation and 2D collection profiles are multiplied to produce sensitivity distribution. The sensitivity profile will be used directly as the intensity information.

By knowing the velocity and intensity of every possible particle flowing through

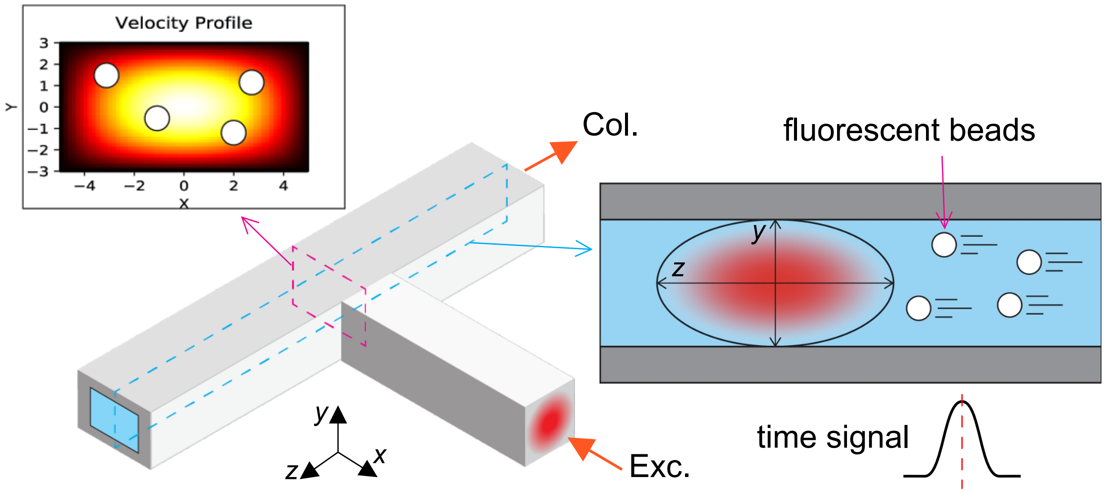


Figure A.3: Simulation model to predict flowing particles event distribution from the APD time trace.

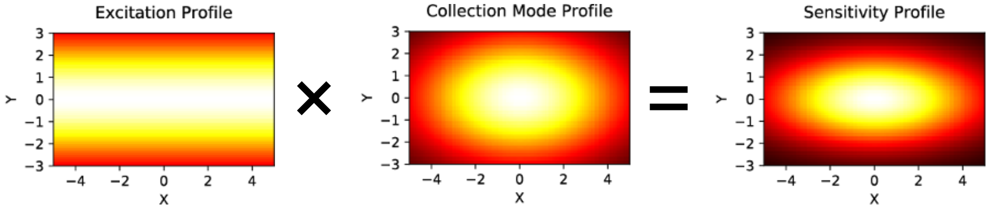


Figure A.4: Sensitivity profile for an ARROW device based on excitation and collection mode profiles.

the channel we can generate a 2D-joint histogram as previously used throughout this thesis. Here, we need to populate events based on where they are positioned inside the channel. For example, if they are closer to the center of the channel, there is a higher chance of having more events collected, because the stream is faster and more particles per time unit is crossing the excitation volume. This helps to add the necessary statistical information to the model. Fig. A.5 shows an example of a simulated model for a chip with corresponding profile information. Here, $ex : y(0, 0.3h)$ means the excitation profile parameters are $\mu = 0, \sigma = 0.3 \times h$. Similarly, horizontal (x) and vertical (y) parameters of the collection profile are described under the plot.

By varying the excitation and collection profile parameters, we can generate a grid of expected distributions as depicted in Fig. A.6. This technique was used in this thesis to fit events distribution from the experimental data into the model by comparing the 2D histograms, and gives the insight about possible errors from the fabricated chip without FIBing and destroying the chip. Ideally, we want the event to sit in the top-left corner of the 2D histogram where the intensity and velocity have no variations and have the maximum values.

The following code includes the entire algorithm to calculate distributions, overlap the profiles, and generate the plots.

```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

h,w = 6,12  # height and width of channel
dy, dx = .1, .1 # y, x resolution
Y = np.arange(-h/2,h/2+dy,dy)
```

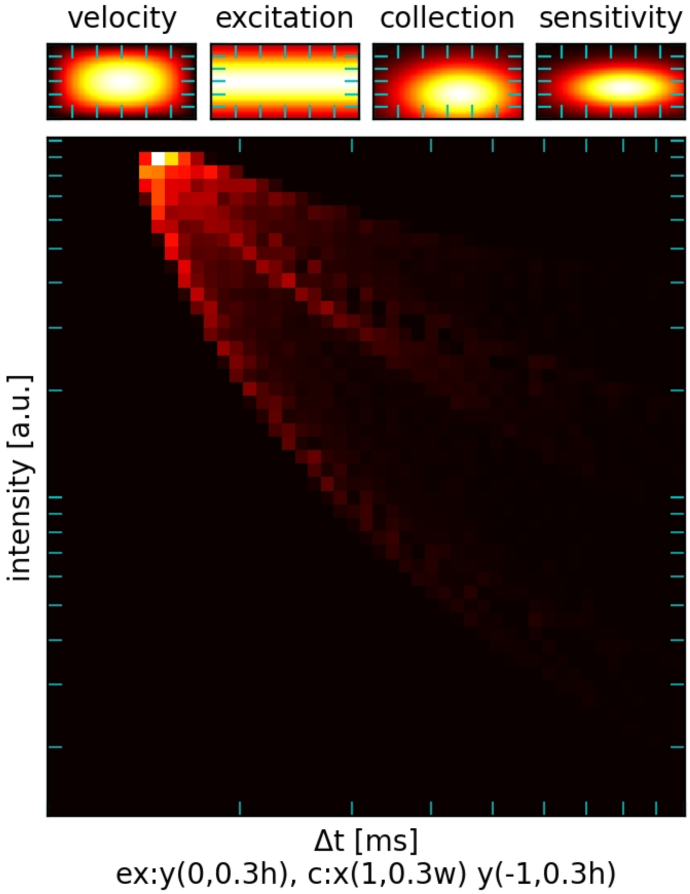


Figure A.5: Simulated event distribution for arbitrary excitation and collection distributions.

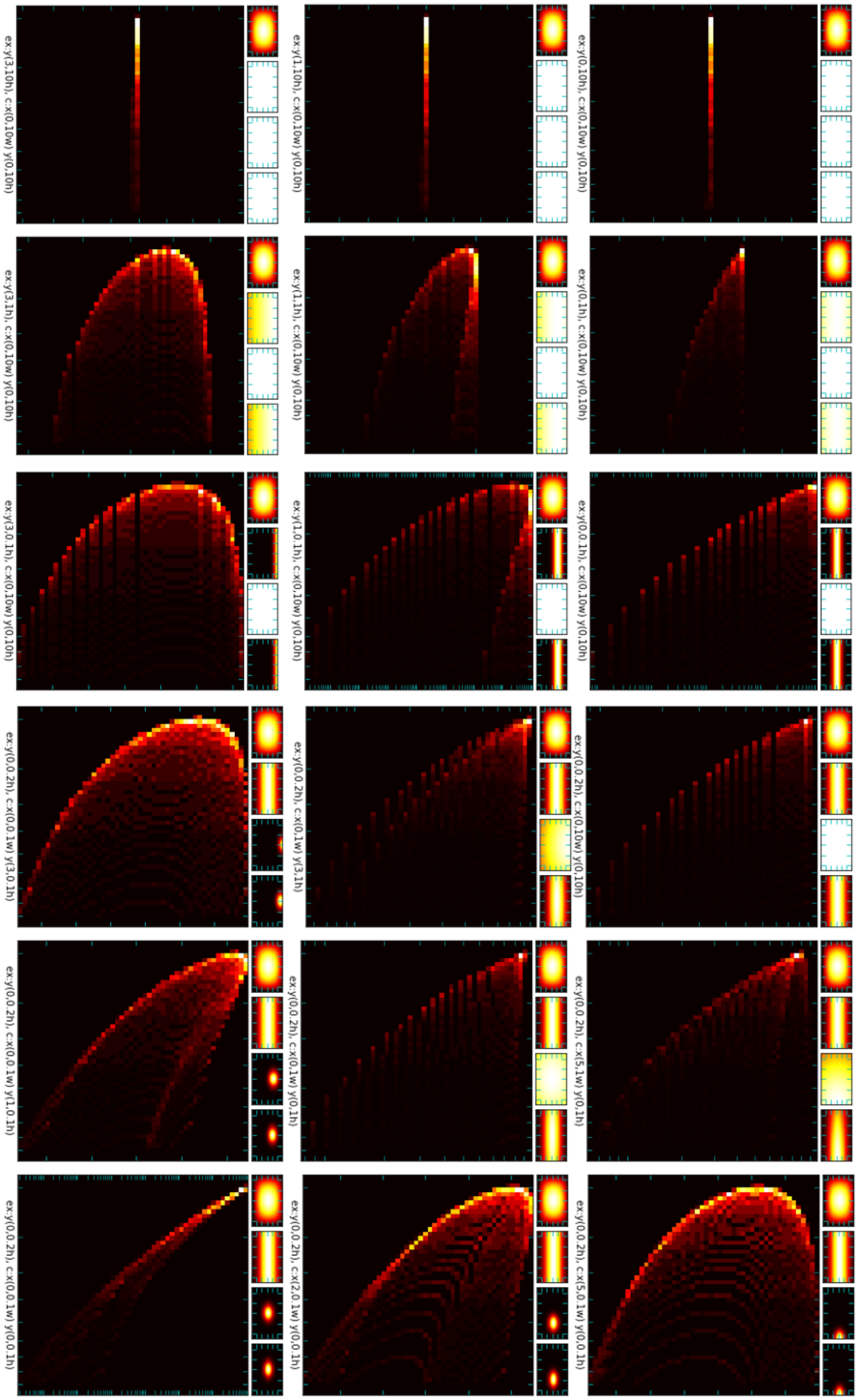


Figure A.6: Excitation and collection profile variations effect on event distribution.

```python
X = np.arange(-w/2,w/2+dx,dx)
XX, YY = np.meshgrid(X,Y)   # create x-y (channel cross-section) plane
repeat = 10     # simulates longer experiment (xN events per pixel)
bins = 50       # histogram bins
G = 7           # Parabolic dist. constant
ex_mu, ex_s = 0, 0.3  # exc. profile params (mu, sigma)
c_mux, c_sx, c_muy, c_sy = 1, 0.3, -1, 0.3
↪  # col. profile params (mu_x, sigma_x, mu_y, sigma_y)
ex_modes = [[ex_mu,ex_s*h]] # single-mode exc.
# ex_modes = [[-h/4,h/4],[h/4,h/4]] # double-lobe mode exc.
U_profile = G*(1-((YY-0)/h*2)**2)*(1-((XX-0)/w*2)**2)
# uncomment to simulate 3D-HDF effect
# (limit locations a bead can be present by cropping velocity dist)
# U_profile[np.where(XX<-2)] = 0
# U_profile[np.where(XX>2)] = 0
# U_profile[np.where(YY<-1.5)] = 0
# U_profile[np.where(YY>1.5)] = 0
Ex_profile = 0
for ex_mode in ex_modes:
  Ex_profile += np.exp(-0.5*((YY+ex_mode[0])/ex_mode[1])**2)
c_modes = [[c_muy,c_sy*h,c_mux,c_sx*w]] # single-mode col.
C_profile = 0
for c_mode in c_modes:
  C_profile += np.exp(-0.5*((YY+c_mode[0])/c_mode[1])**2-0.5*((XX-c_mode[2])/c_mode[
  ↪  3])**2)
Sens_profile = np.multiply(C_profile,Ex_profile) # sensitivity dist.
# create figure and configure subplots
fig = plt.figure(figsize=(4,5),dpi=200)
gs_summaryplot = fig.add_gridspec(2, 4, height_ratios=[1,6])
ax = [0]*5
ax[0] = fig.add_subplot(gs_summaryplot[0,0])
ax[1] = fig.add_subplot(gs_summaryplot[0,1])
ax[2] = fig.add_subplot(gs_summaryplot[0,2])
ax[3] = fig.add_subplot(gs_summaryplot[0,3])
ax[4] = fig.add_subplot(gs_summaryplot[1,:])
plt.subplots_adjust(wspace=0.1,hspace=0.00)
# show velocity dist.
ax[0].matshow(U_profile,extent=[X[0],X[-1],Y[0],Y[-1]],cmap='hot',vmin=0)
ax[0].tick_params(axis='both',direction='in',which='both',color='c',bottom=True,
↪  right=True,labelleft=False,labeltop=False, length=5)
ax[0].set_title("velocity",fontsize=10)
# show exc. dist.
ax[1].matshow(Ex_profile,extent=[X[0],X[-1],Y[0],Y[-1]],cmap='hot',vmin=0)
```

```python
ax[1].tick_params(axis='both',direction='in',which='both',color='c',bottom=True,right=True,labelleft=False,labeltop=False, length=5)
ax[1].set_title("excitation",fontsize=10)
# show col. dist
ax[2].matshow(C_profile,extent=[X[0],X[-1],Y[0],Y[-1]],cmap='hot',vmin=0)
ax[2].tick_params(axis='both',direction='in',which='both',color='c',bottom=True,right=True,labelleft=False,labeltop=False, length=5)
ax[2].set_title("collection",fontsize=10)
# show sens. dist.
ax[3].matshow(Sens_profile,extent=[X[0],X[-1],Y[0],Y[-1]],cmap='hot',vmin=0)
ax[3].tick_params(axis='both',direction='in',which='both',color='c',bottom=True,right=True,labelleft=False,labeltop=False, length=5)
ax[3].set_title("sensitivity",fontsize=10)
# convert velo. to 'dt (scale)' and use sens. value as intensity of time signal
df = pd.DataFrame({'scale':U_profile.flatten()**(-1), 'coeff':Sens_profile.flatten()})
# probability is proportional to velocity of a particle
# determines how many events can be expected to come from a certain location of channel
df['probability'] = 1/df['scale']
df['probability'] = df['probability']/df['probability'].max() # normalize
# repeat events from high velocity locations to have
# more events of that part of stream in the dataset
df = df.iloc[np.array([range(len(df))]).T.repeat(df['probability']*repeat).flatten()]
# prepare logscale axis values for 2D-hist (similar to PCWA analysis in Ch3)
xlogbins = np.logspace(np.log10(0.1),np.log10(1.0),bins)
hist, ybins = np.histogram(df['coeff'],bins)
ylogbins = np.logspace(np.log10(0.9*ybins[0]),np.log10(1.1*ybins[-1]),len(ybins))
ax[4].hist2d(df['scale'],df['coeff'],cmap='hot',bins=[xlogbins, ylogbins])
ax[4].loglog()
ax[4].tick_params(axis='both',direction='in',which='both',color='c',top=True,right=True,labelleft=False,labelbottom=False, length=5)
ax[4].set_ylabel("intensity [a.u.]")
ax[4].set_xlabel(f"\u0394t [ms]\nex:y({ex_mu:.2g},{ex_s:.2g}h), c:x({c_mux:.2g},{c_sx:.2g}w) y({c_muy:.2g},{c_sy:.2g}h)")
plt.show()
```