An Artificial Neural Network-based Signal Classifier for Automated Identification of Detection Signals from a Dielectrophoretic Cytometer

by

Ashlesha Bhide

A Thesis submitted to the Faculty of Graduate Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering

University of Manitoba

Winnipeg

Abstract

An automated signal classifier and a semi-automated signal identifier are designed for collecting the dielectrophoretic signatures of cells flowing through a dielectrophoretic cytometer. The label-free dielectrophoresis (*DEP*) cytometer is capable of independently detecting and actuating biological cells. Electronic detection signals generated by the cells are related to the Claussius-Mossotti factor (CMF) and the altitude of cells in the channel. This allows the dielectric properties of cells to be connected to their physiological properties. In past work, the *DEP* cytometer signals were manually sorted by going through all recorded signals, which is impractical when analyzing 1000's of cells per day. In the semi-automated method of collection, signals are automatically identified as events and displayed on the user interface to be accepted or rejected by the user. This approach reduced signal collection time by more than half and produced statistics nearly identical to the manual method. The automated signal classifier based on pattern recognition categorizes detection signals as 'Accept' or 'Reject'. Classifying detection signals requires training a multilayered feed-forward network with an algorithm to optimize the performance of the network. The classifier is robust in accurately classifying detection signals without altering the statistics compared to those generated by manual and semi-automated data collection methods. Analyzing large volumes of detection signals is possible in much reduced times and may be approaching real time capability.

Acknowledgements

First of all, I wish to sincerely thank my advisor Dr. Douglas. J. Thomson for giving me an excellent opportunity to contribute to this project, and for guiding me in the right direction. I am very grateful to him for being patient, kind and encouraging throughout the M.Sc. program. I hope to have stood up to his expectations. I wish to express my sincere gratitude to Dr. Gregory. E. Bridges for all his valuable suggestions and feedback.

I would like to thank my committee members Dr. Dean McNeill and Dr. Jitendra Paliwal for evaluating the research work presented in this thesis and for providing a constructive feedback.

I appreciate all the support and assistance received from Dr. Marija Nikolic-Jaric, Tim Cabel, Elham Salimi, Kaveh Mohammed, and Dr. Graham Ferrier. I wish to acknowledge Bahareh Saboktakin Rizi for her assistance and equal contribution in designing the Semi-Automated Data collection system. I will cherish all the conversations and useful discussions I had with Sharmishtha Bhadra, Khalada Parveen Tuli, Dilruba Jeba Zaman, and all other friends.

I would like to acknowledge the financial support provided by the Natural Sciences and Engineering Research Council (NSERC), the Canada Foundation for Innovation (CFI), Western Economic Diversification Canada (WD), Canadian Microelectronics Corporation (CMC) Microsystems, and the Province of Manitoba for this research. I would like to thank the Faculty of Graduate Studies, Department of Electrical and Computer Engineering, and the Dean's Office for providing financial support to present my work at I^2MTC.

Last and not the least, I dedicate this thesis to my wonderful parents for their unconditional love, endless support, and for motivating me to always learn. I owe my deepest gratitude to my brother, Ameya, for giving me the courage to pursue my dreams and for helping me make the correct choices. I thank Lord Almighty for all the blessings.

Table of Contents

List of Tables

List of Figures

List of Copyrighted Material for which permission was obtained

Figures 4.2, 4.3 and 4.8 in this thesis are drawn based on Figures on pages 1- 4, 1- 10, and 1- 11 respectively from the user manual, "Neural Network Toolbox™ User's Guide", written by M. H. Beale, M. T. Hagan, and H. B. Demuth with kind permission from The MathWorks, Inc.

Nomenclature

Abbreviations

DEP Dielectrophoresis or Dielectrophoretic

CMF Clausius-Mossotti Factor

CHO Chinese Hamster Ovary

nDEP Negative Dielectrophoresis or Negative Dielectrophoretic

pDEP Positive Dielectrophoresis or Positive Dielectrophoretic

ANN Artificial Neural Network

AN Artificial Neuron

MSE Mean Squared Error

Constants

$\varepsilon_0 = 8.854 \times 10^{-12}$ F/m Permittivity of free space

Chapter 1

Background

1.1 Introduction

The mechanisms underlying the regular growth and development of biological cells can be investigated by single-cell analysis. This analysis is also important in understanding the reasons for malfunction in intracellular processes leading to diseases [1] [2]. Dielectrophoresis (*DEP*) is a convenient method used to analyze and characterize single biological cells [3] [4]. *DEP* is a phenomenon in which a force is exerted on a dielectric particle when subjected to non-uniform electric fields [5]. *DEP*-based techniques have been used successfully over the years to characterize, sort, isolate, separate and manipulate biological cells depending on their dielectric properties [6] [7]. Cell processes are usually tracked by incubating cells with specific dyes to reveal variations in their properties [8] [9] [10]; however there is a possibility that the dyes themselves may cause a change in the cell itself. Electronic single cell analysis methods based on *DEP* are label-free and have been used to separate viable cells from non-viable yeast cells [11] [12]; separate cancerous cells from non-cancerous ones [13]; separate different cell types in blood analysis [14]; identify different types of tumor cells [15]; and to study stages of apoptosis in leukemia cells [16].

Cytometers are used to analyze single biological cells to classify their state [8]. Any abnormality in the healthy functioning of a cell is often an indication of a disease.

Healthy (viable) cells and non-viable (cancerous/apoptotic) cells can often exhibit similar morphology. However, the variations in the state of cells are often associated with changes in their dielectric properties, which are directly related to their physiology [17] [18]. Electronic microfluidic devices offer an easy method to determine the dielectric properties of the cells [19]. It is possible to analyze large number of biological cells and obtain a precise description of the dielectric properties of a population of cells [20] [21]. The polarizability of a cell is related to its composition by a quantity known as Claussius-Mossotti factor (CMF). CMF variation over frequency has information about physiological and dielectric properties of the cells.

An all-electronic detection and actuation sensor based on *DEP* cytometry developed at University of Manitoba makes use of *DEP* to monitor the physiological properties of cells [22]. The all-electronic sensor is sensitive to the changes in the physiological properties of the cell and produces electronic detection signatures. These signatures arise from the *DEP* actuation of cells and are explained in detail in Section 2.1. The electronic sensor is label-free; non-invasive; it allows analysis of large populations of cells (1000's); it can be miniaturized; it allows a higher degree of electronic integration; and it is suitable for low-cost applications in research and healthcare [20] [21]. The application of this sensor is to track the onset of apoptosis in CHO cells actuated by *DEP* voltages at 6MHz frequency which is studied and established in [21].

Apoptosis is programmed cell death related to the physiological changes occurring within multicellular organisms. In early apoptotic stages, morphologically, the cell shrinks and its chromatin condenses. Cell shrinkage is associated with changes in the ionic composition of the cell's cytoplasm; the conductivity of the cytoplasm drops. The

polarizability of the cell changes, thereby, influencing its response to *DEP* actuation [23] [24].

1.2 Detection

The interior of a cell is composed of ions and charged molecules. Under the influence of a non-uniform electric field, these ions get charged and orient themselves with respect to the electric field. Changes in the polarizability of the cell can be a result of the physiological changes occurring within a cell causing changes in the concentration of ions and the morphology of the cell [25]. The presence of a cell in the microfluidic channel is detected by measuring the change in capacitance, ΔC, induced by the cell as it flows over the detection electrodes. When the cell flows over the detection electrodes, it displaces the fluid medium in which it is suspended with the dielectric property of the cell causing a change in capacitance. The apparatus and the interferometer used to detect ΔC (F) are explained in detail in [22].

$$\Delta C = 3\varepsilon_m V_{cell} Re\{K_{CM}\} \frac{E_{rms}^2}{U_{rms}^2} \tag{1.1}$$

where ε_m (F/m) is the real part of the dielectric permittivity of fluid medium, $V_{cell} = \frac{4}{3}\pi R^3$ is the volume of the cell, E_{rms} (V/m) and U_{rms} (V) are the root-mean-squared values of the electric field at the center of the cell and the electric voltage applied to the electrodes, respectively, and $Re\{K_{CM}\}$ represents the real part of the Clausius-Mossotti factor as expressed in Equation 1.2.

1.3 Clausius-Mossotti Factor

The Clausius-Mossotti factor is the measure of the cell's polarizability with respect to its surrounding fluid medium. At a frequency of ω (rad/sec), the Claussius-Mossotti factor K_{CM} is related to the complex dielectric permittivities ($\tilde{\varepsilon}_p$,$\tilde{\varepsilon}_m$) and the conductivities ($\tilde{\sigma}_p, \tilde{\sigma}_m$) of the cell and the medium in which the cell is suspended.

$$K_{CM} = \frac{\tilde{\varepsilon}_p - \tilde{\varepsilon}_m}{\tilde{\varepsilon}_p + 2\tilde{\varepsilon}_m} \tag{1.2}$$

where $$\tilde{\varepsilon} = \varepsilon - j\frac{\sigma}{\omega} \tag{1.3}$$

Since $\tilde{\varepsilon}$ is frequency dependent, the cell responds differently to the electric fields at different frequencies. The real part of the Clausius-Mossotti factor $Re\{K_{CM}\}$ contains the frequency dependence and the polarity of the *DEP* force. $Re\{K_{CM}\}$ for a spherical cell is expressed in [5] as

$$Re\{K_{CM}\} = K_{inf} + \left(\frac{K_0 - K_{inf}}{1 + \omega^2 \tau_{MW}^2}\right) \tag{1.4}$$

where $K_{inf} = (\varepsilon_p - \varepsilon_m)/(\varepsilon_p + 2\varepsilon_m)$, $K_0 = (\sigma_p - \sigma_m)/(\sigma_p + 2\sigma_m)$, and the relaxation time $\tau_{MW} = (\varepsilon_p + 2\varepsilon_m)/(\sigma_p + 2\sigma_m)$.

At low frequencies, the cell response is mainly due to the conductivities of the cell and the medium. The cell response is dominated by the dielectric permittivities of the cell and the suspending medium at higher frequencies [20].

1.4 Actuation

The actuation of a cell is dependent on the dielectrophoretic force F_{DEP} (N). The cell, when subjected to a non-uniform electric field, experiences a force that is directed along the gradient of the electric field, and is expressed in [5] as

$$F_{DEP} = \frac{3}{2}\varepsilon_m V_{cell} Re\{K_{CM}\}\nabla(E_{rms}^2) \tag{1.5}$$

The polarity of the CMF indicates whether *DEP* force is acting along or against the gradient of the electric field $\nabla(E_{rms}^2)$ (V^2/m^3), which is termed as positive *DEP* force (*pDEP*) or negative *DEP* (*nDEP*) force respectively. A cell remains unactuated when there is no *DEP* force acting on a cell; the magnitude of electric field gradient is zero [5] [26]. The variation of $Re\{K_{CM}\}$ with frequency for a viable and a non-viable CHO cell suspended in a medium of conductivity 0.17S/m is shown in Figure 1.1. $Re\{K_{CM}\}$ at each frequency point is calculated from Equation 1.4. A viable CHO cell experiences different forces in different frequency ranges while a non-viable CHO cell experiences a negative *DEP* force throughout the entire frequency range. At 0.1MHz, a viable CHO cell experiences a negative *DEP* force while at 6MHz, its experiences a positive *DEP* force. The crossover frequency at which a viable CHO cell begins to experience *pDEP* is ≈0.5MHz. A non-viable (apoptotic) cell experiences negative *DEP* at both 0.1MHz and 6MHz.

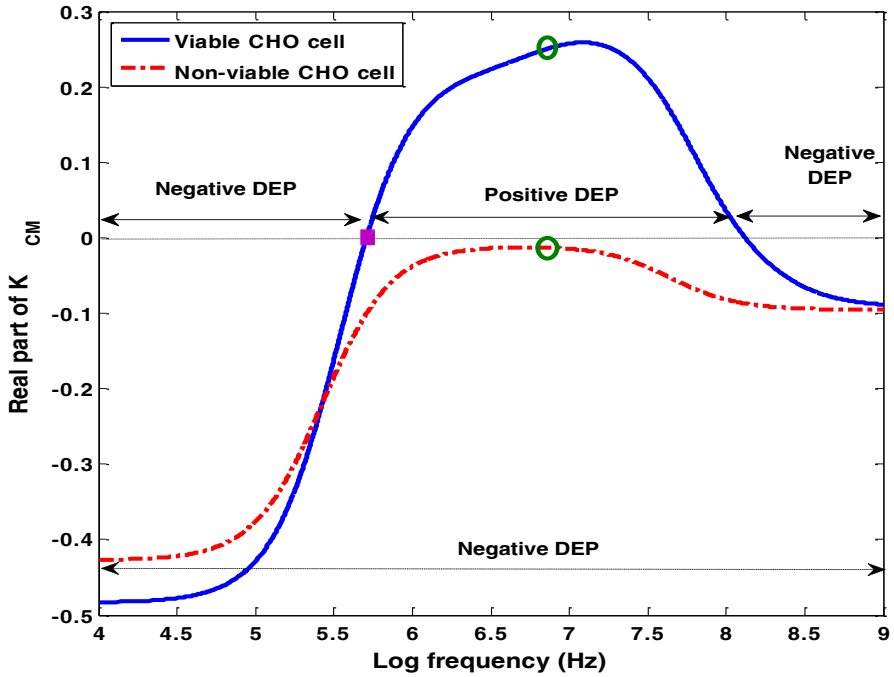


Figure 1.1: $Re\{K_{CM}\}$ vs. Frequency for viable and non-viable CHO cells. 'o' indicates the dielectric response of viable and non-viable CHO cells at 6MHz actuation frequency.'□' indicates the crossover frequency (≈0.5MHz) at which the viable cells begin to experience *pDEP* force.

1.5 Electrorotation and Impedance Cytometry

Apart from *DEP* cytometry, two other dielectric based cell analysis techniques used are: Electrorotation and Impedance cytometry.

A biological cell, subjected to a rotating electric field, experiences a torque causing it to rotate. The torque Γ (N/m) on the cell is due the phase difference between the applied rotating field and the induced dipole and is expressed in [25] as

$$\Gamma(\omega) = -4\pi\varepsilon_m R^3 Im\{K_{CM}\}E^2 \tag{1.6}$$

where R is the radius of the cell, ε_m is the dielectric permittivity of the fluid medium, $Im\{K_{CM}\}$is the imaginary part of CMF, and E is the magnitude of the electric field. The frequency dependence of the torque is presented by $Im\{K_{CM}\}$; the cell rotates along or against the electric field when the polarity of $Im\{K_{CM}\}$ is negative or positive respectively. The rate of rotation (rad/sec) of the cell and viscosity of the medium in which it rotates can be used to estimate $Im\{K_{CM}\}$ [25]. The electrorotation spectrum yields the necessary information about the dielectric properties of the cell and can be used to distinguish between different cells. The limitations of this analysis technique are: 1) One electrorotation measurement requires a few seconds per cell limiting the number of cells analyzed; and 2) Observing rotations of smaller cells is very challenging [25] [26].

In Impedance cytometry, the dielectric properties of biological cells are determined by exciting them with a frequency-dependent voltage signal and measuring its current response. According to Maxwell's mixture equation [27], the complex permittivity for a mixture $\tilde{\varepsilon}_{mix}$ of biological cells in the suspending medium is expressed in [28] as

$$\tilde{\varepsilon}_{mix} = \tilde{\varepsilon}_m \frac{1+2\varphi K_{CM}(\omega)}{1-\varphi K_{CM}(\omega)} \approx \tilde{\varepsilon}_m.(1+3\varphi K_{CM}(\omega)) \tag{1.7}$$

where $\tilde{\varepsilon}_m$ is the permittivity of the medium, φ is the volume fraction occupied by the cells in the entire volume. The impedance of the mixture Z_{mix} is related to its complex permittivity and is expressed in [28] as

$$Z_{mix} = \frac{k}{j\omega\tilde{\varepsilon}_{mix}} \approx \frac{k}{j\omega\tilde{\varepsilon}_m}(1-3\varphi K_{CM}(\omega)) \tag{1.8}$$

k is the geometric cell constant. Impedance cytometry is a convenient and a faster method of obtaining the bulk dielectric properties of a mixture of cells in a suspending medium.

The next chapter discusses the microfluidic device used to independently detect and actuate single biological cells. The change in altitude of the cell between the two detection sites, as a result of *DEP* actuation, is related to the polarizability of the cell. Experimental and simulated detection signals generated by biological cells from different types of actuations are shown. The change in the dielectric properties of the cell modifies the electronic detection signals and can be connected to their physiology by a quantity called the force index.

Chapter 2

Experimental setup: Detection and DEP Actuation of Biological Cells

2.1 Microfluidic device for simultaneous detection and DEP Actuation of biological cells

A microfluidic device along with an RF interferometer allows us to study the dielectric properties of biological cells by simultaneous detection and actuation [21]. The microfluidic device, shown in Figure 2.1(a), consists of an H-shaped microfluidic channel with a differential coplanar electrode array fabricated at the bottom of the channel. The *H*-shaped channel has two inlet ports (*I1* and *I3*) and two outlet ports (*I2* and *I4*). The channel height, width and length are 40, 120 and 500μm respectively. A micrograph of the electrode configuration is shown in Figure 2.1(b). Each electrode is 25μm wide, the spacing between the two central electrodes *A* is 15μm, the gap between the electrode sets *D1* and *D2* is 25μm, and the spacing between the electrodes *A* and electrodes *D1* and *D2* is 35μm on each side. A sample of biological cells suspended in a low conductivity medium is pumped through the microfluidic channel. Regulating the pressure applied to the ports controls the direction and the flow velocity of the cells.

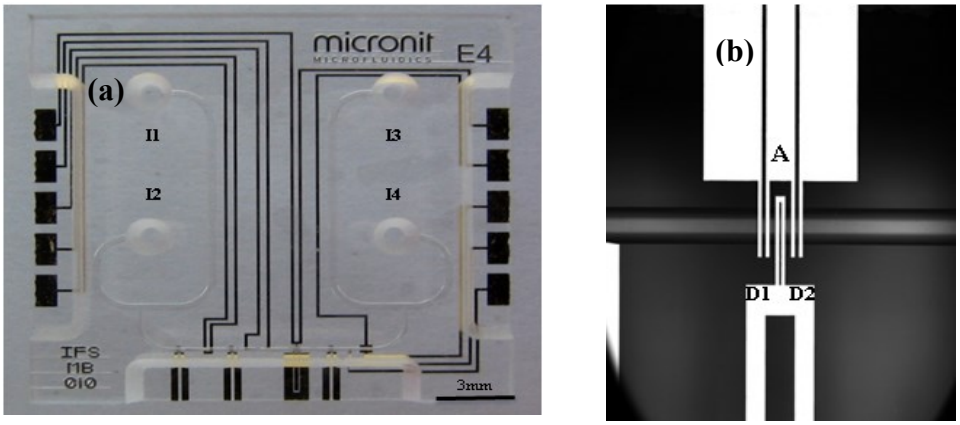


Figure 2.1: (a) The microfluidic device with a microfluidic channel, four ports (*I1, I2, I3* and *I4*) and coplanar electrode array fabricated at the bottom of the channel. (b) Micrograph of the coplanar electrode configuration.

A cell flowing over the electrode array is detected by the outer detection electrodes (*D1* and *D2*) and actuated by the inner electrodes (*A*). The detection electrodes are energized by a gigahertz frequency (1.45GHz) signal of amplitude less than $300mV_{p\text{-}p}$; the actuation electrodes are energized by a megahertz signal of amplitude $4\text{-}8V_{p\text{-}p}$ [21]. The coplanar electrode array generates non-uniform electric fields in the region above the electrodes. The actuation electrodes (*A*) apply a dielectrophoretic force to the cell resulting in a change in the altitude of the cell. The actuation can be monitored by examining the changes in cell altitudes before and after actuation. The presence of a cell is detected by a signal *S* which is proportional to the change in capacitance ΔC (Equation 1.1) induced by the cell between the detection electrodes. The detection signal *S* is also proportional to E^2_{rms}, hence, it also depends on the measured altitude of the cell as shown in the simulation result shown in Figure 2.2 [20].

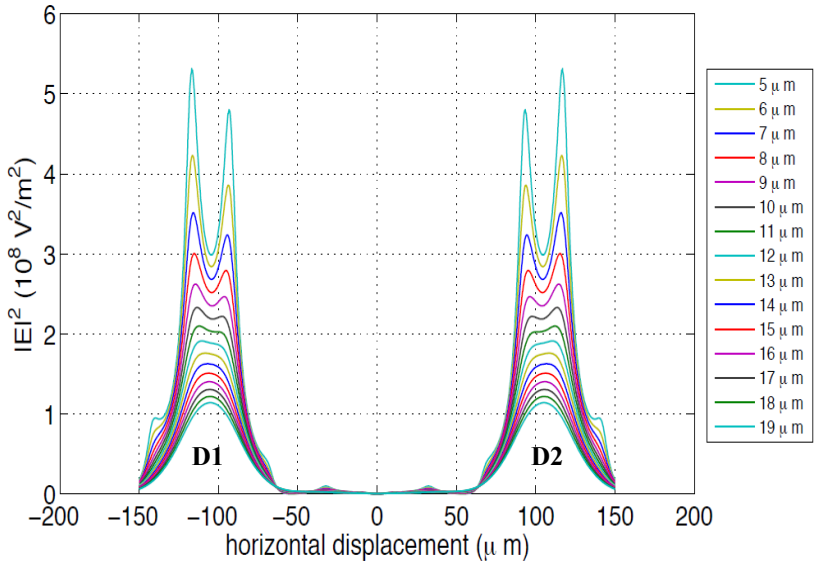


Figure 2.2: E^2_{rms} simulated for a cell at different altitudes in the channel. The altitude of the cell at *D2* is affected by actuation. Simulations were carried out in COMSOL Multiphysics® for a 13μm- diameter CHO cell flowing over the electrode array at different constant elevations from 5- 19μm.

Initially, a cell entering the channel is detected by the electrodes *D1* resulting in signal amplitude P_1. As the cell flows over electrodes *A*, it is actuated by a *DEP* force which causes the cell to change its altitude. After actuation, the cell is detected at its changed altitude, by electrodes *D2* resulting in a signal amplitude P_2. When the actuation electrodes are energized by a low frequency signal between 0.1-20MHz, the generated electric field exerts a *DEP* force on the cell. If the force acts in the direction of the field gradient, the cell is attracted to the actuation electrodes. Conversely, if the force acts in the direction against the field gradient, the cell is repelled by the actuation electrodes. The *DEP* actuation force is determined by the polarizability of the cell in its surrounding medium and the actuation frequency.

A 13μm-diameter viable CHO cell ($\varepsilon_p = 6.8\varepsilon_0$ and $\sigma_p = 3e^{-6}$S/m) [29] [30] suspended in a low conductivity medium ($\varepsilon_m = 78\varepsilon_0$ and $\sigma_m = 0.17$S/m), when actuated at 6MHz, is attracted to the electrodes and is said to have experienced a *pDEP* force (Figure 2.3). At the same actuation frequency, a non-viable 13μm-diameter viable CHO cell is pushed away by the actuation electrodes to the weaker field regions and is said to have experienced *nDEP* force (Figure 2.3). Experimentally obtained detection signals (*S*) for a viable and a non-viable CHO cell experiencing *pDEP* and *nDEP* force, respectively, are shown in Figures 2.4(a) and 2.4(b). When no *DEP* actuation voltage is applied, the altitude of the cell remains unaltered. An unactuated CHO cell and its corresponding detection signal (*S*) are shown in Figures 2.5 and 2.6 respectively.

A parameter called force index 'Φ' quantifies the change in the cell signal *S* as a result of *DEP* actuation. Force Index is represented as a normalized ratio of the difference and sum of signal peak amplitudes before (P_1) and after actuation (P_2), and is expressed in Equation 2.1.

$$\Phi = \frac{2(P_2 - P_1)}{P_2 + P_1} \tag{2.1}$$

For positive and negative Φ, the cell is said to have been positively (*pDEP*) and negatively (*nDEP*) actuated respectively. No actuation corresponds to Φ=0. Force index value corresponds to the strength of the *DEP* force that causes a change in the altitude of the cell. Theoretically, Φ lies between -2 and +2. The signal for a *pDEP* experiencing cell shows that $P_1 < P_2$; the signal for an *nDEP* experiencing cell shows that $P_1 > P_2$; the signal for an unactuated cell shows that $P_1 = P_2$.

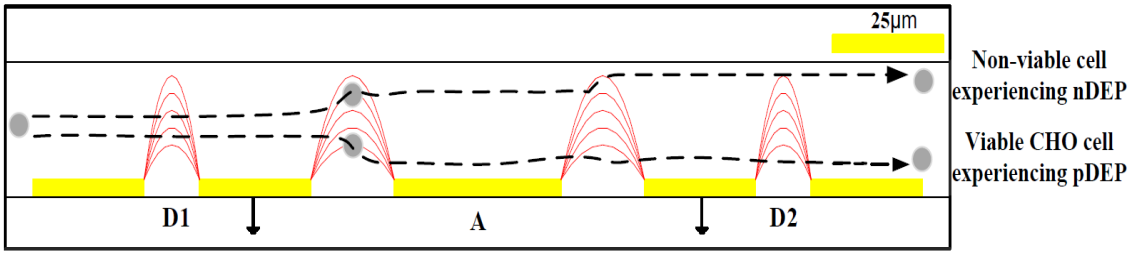


Figure 2.3: Side view of the channel depicting typical cell trajectories traced by CHO cells when actuated by a 6MHz frequency signal.

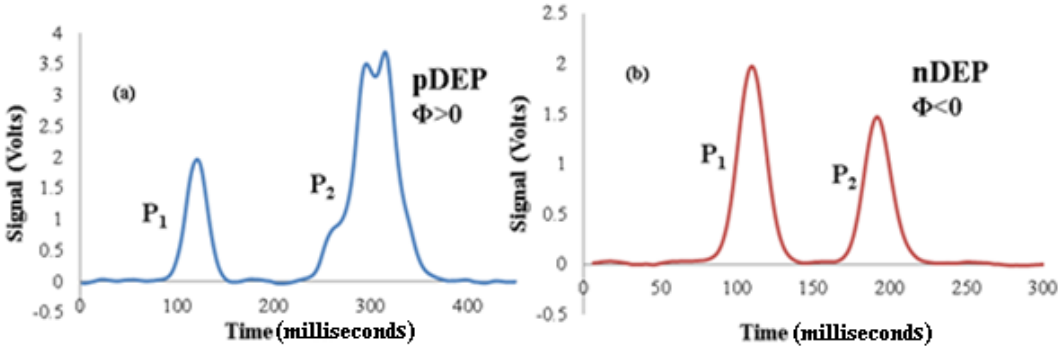


Figure 2.4: (a) Experimental detection signal for a viable CHO cell experiencing *pDEP*; $P_1<P_2$; and Φ>0. (b) Experimental detection signal for a non-viable CHO cell experiencing n*DEP*; $P_1> P_2$; and Φ<0.

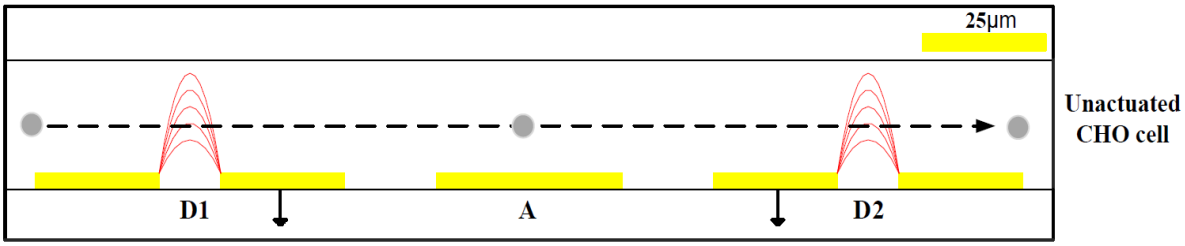


Figure 2.5: Side view of the channel depicting a typical cell trajectory traced by an unactuated CHO cell.

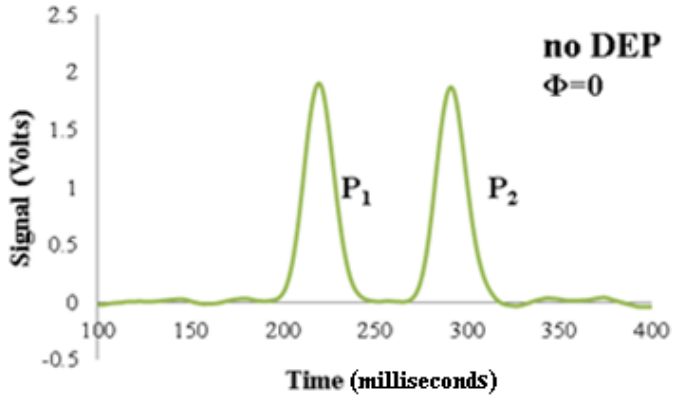


Figure 2.6: Experimental detection signal for an unactuated CHO cell; P_1=P_2; and Φ=0.

2.2 Simulation results of detection signals and trajectories

Simulations (2-D) were performed on a model cell using COMSOL Multiphysics® to compare with experimental results. All simulations are done for a cell of diameter 13μm and actuation voltage of $4V_{p\text{-}p}$ at 6MHz frequency. The altitude of the cell is the distance from the bottom of the channel to the center of mass of the cell. The altitude of a cell in the channel can be related to the actuation force (directly related to the polarity of CMF) experienced by the cell. Simulated detection signals of CHO cells for different types of actuations are shown in Figure 2.7. At detection site *D2*, the peak amplitude diminishes as the cell is pushed up to higher altitudes in the channel as shown in Figure 2.8. A cell detected at site *D1* is 12μm high in the channel: (1) if positively actuated (Φ>0), the cell is attracted to the actuation electrodes and is detected at site *D2* at altitudes lower than 12μm; (2) if unactuated (Φ=0), the cell is detected at *D2* at 12μm; (3) if negatively actuated (Φ<0), the cell is repelled by the actuation electrodes and is detected at site *D2* at altitudes higher than 12μm.

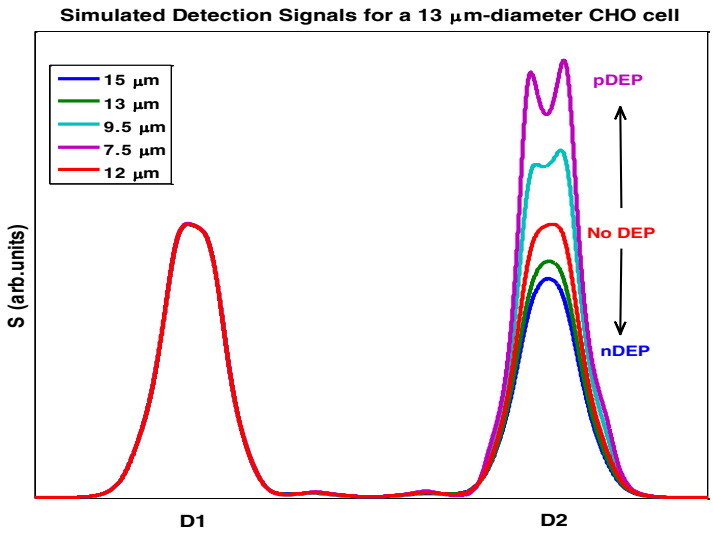


Figure 2.7: Simulated detection signals of a 13µm CHO cell for different types of actuations. The cell enters the channel at constant elevation and after actuation, it moves to a new altitude depending on the type of *DEP* force it experiences.

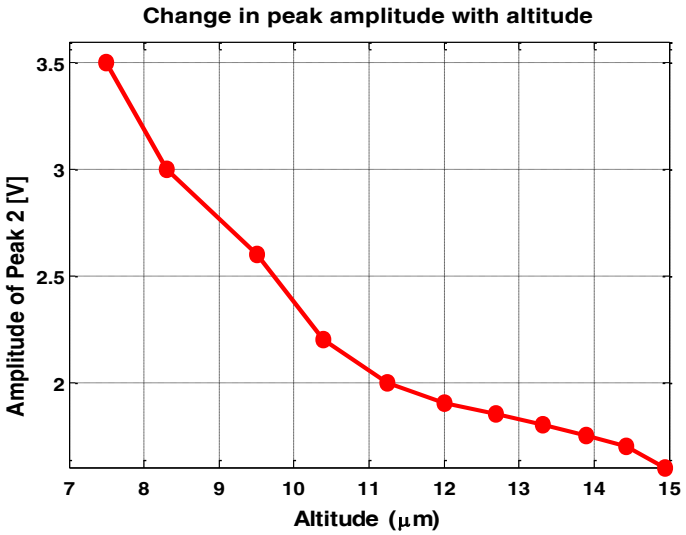


Figure 2.8: Change in peak amplitude with altitude.

The cell altitudes at detection sites *D1* and *D2* are marked as h_1 and h_2 respectively; the change is cell altitude (h_1-h_2) is denoted by Δh. For a *pDEP* signal, the change in cell altitude is positive ($h_1 > h_2$) as the cell is pulled to a lower altitude. For an unactuated

signal, there is no change in cell altitude (h_1=h_2). For *nDEP* signal, the change in cell altitude is negative (h_2>h_1) as the cell is pushed to a higher altitude. The change in cell altitude Δh with change in CMF is shown in Figure 2.9.

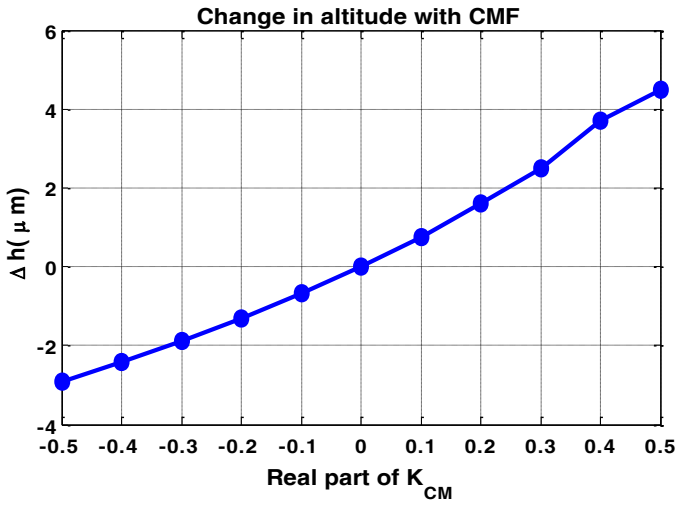


Figure 2.9: Change in altitude Δh with $Re\{K_{CM}\}$.

The motion of a cell in the channel is governed by the gravitational, hydrodynamic lift, *DEP* actuation, and drag forces acting on it [20]. The fluid flow in the channel has a parabolic profile and the parabolic flow velocity 'v_m' is expressed as

$$v_m = 6 < v > \frac{h}{H}\left(1 - \frac{h}{H}\right) \tag{2.2}$$

where <v> is the average fluid flow velocity, H is the height of the channel, and h is the distance from the bottom of the channel. The maximum fluid velocity is in the centre of the channel and it is zero at the walls of the channel. The modelled trajectories are calculated by solving Equation 2.3 [20].

$$F = F_{DEP} + F_{drag} + F_{gravity} + F_{lift} \tag{2.3}$$

The *DEP* force is calculated from Equation 1.5. The fluid drag force F_{drag} is expressed as

$$F_{drag} = 6\pi\eta R(v_p - v_m) \tag{2.4}$$

where η is the viscosity of the fluid medium, R is the radius of the cell, and v_p and v_m are the velocities of the cell and the medium respectively. The gravitational force, counter-balanced by the buoyant force, is expressed as

$$F_{gravity} = \frac{4\pi R^3}{3} g(\rho_p - \rho_m) \tag{2.5}$$

where g is the acceleration due to gravity (9.81ms^{-2}), and ρ_p and ρ_m are the densities of the cell and medium respectively. ρ_p = 1050kgm^{-3} and ρ_m = 1000kgm^{-3}.

The hydrodynamic lift force F_{lift} is expressed as

$$F_{lift} = k_l \frac{6\eta<v>R^3}{H(h-R)} \tag{2.6}$$

where k_l is the lift force constant, H = 40μm, R = 6.5μm, η = 10^{-3} Pas.

The trajectories traced by a 13μm CHO cell in the channel as a result of actuation are simulated by varying CMF values in the interval [-0.5, 0.5]. As shown in Figure 2.10, for a negative CMF the cell is pushed higher up in the channel; for a positive CMF the cell is pulled down towards the channel floor. When the cell is not actuated i.e. CMF = 0, it exits the channel at the same altitude at which it entered the channel.

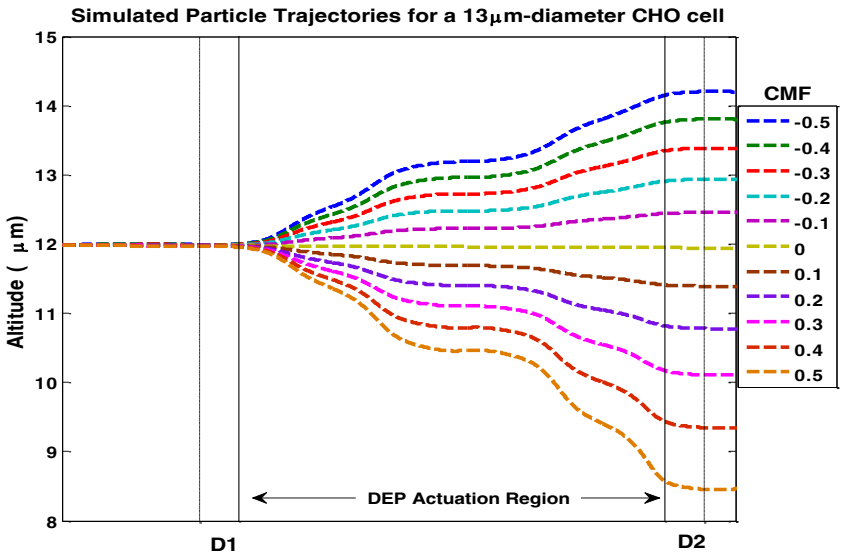


Figure 2.10: Simulated particle trajectories traced by a 13μm CHO cell on varying CMF from -0.5 to 0.5.

The behavior of the device is tested on a population of CHO cells batch cultured in a 3-liter glass bench-top bioreactor at 37˚C and 7.2pH; they are suspended in a low conductivity medium (σ_m = 0.17S/m). All experimental signals, shown in Figures 2.4(a), 2.4(b), and 2.6, have similar peak amplitudes P_1 because the CHO cell sizes are uniform and their flow velocities are adjusted so that they enter the channel at same altitudes. The peak amplitude P_2 depends on the force experienced by CHO cells in the actuation region. The widths of the peaks depend on the velocity of the cell which changes as the cell enters different fluid layers. A *pDEP* experiencing CHO cell signal has a wider peak 2; it slows down when it is attracted to a region with low fluid velocity. Conversely, an *nDEP* experiencing CHO cell has a narrower peak 2 because it is pushed to fluid layers with higher velocities. For an unactuated CHO cell, the cell flows at the same velocity when it enters and exits the channel. This approach allows separation of cells based on

change in their altitudes before and after actuation; it also gives an idea about the cell's physiological state i.e. if cells are viable (healthy) or nonviable.

2.3 Experimental Detection signals of CHO cells

The detection signals are collected in real-time by a data acquisition system (DAQ) as explained in Section 3.1 and analyzed by the event identification programs discussed in Sections 3.3 and 4.10 respectively. Figures 2.11 and 2.12 show unactuated and *DEP*-actuated signals, respectively, obtained in real-time. The positions of the cell in the channel for different types of actuation are depicted in Figures 2.3 and 2.5. Signals shown below yield some information regarding the state of the cell and are accepted by the user during signal analysis.

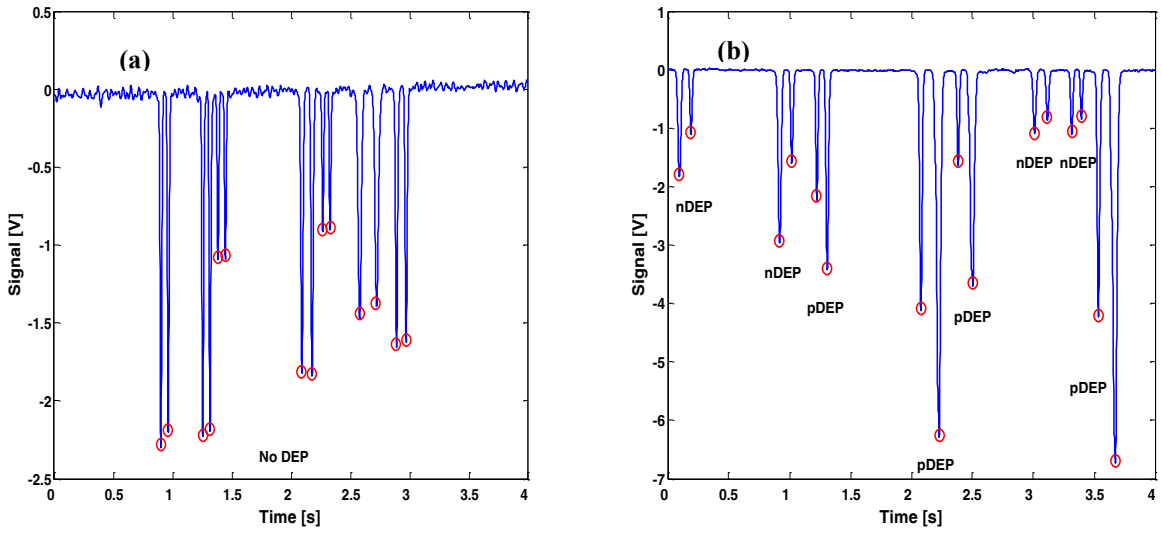


Figure 2.11: (a) Experimentally obtained detection signals for unactuated CHO cells. (b) Experimentally obtained detection signals for positively and negatively actuated CHO cells.

The important features of acceptable signals (*pDEP, nDEP,* and *no DEP*) from which useful information can be extracted are shown in Figure 2.12.

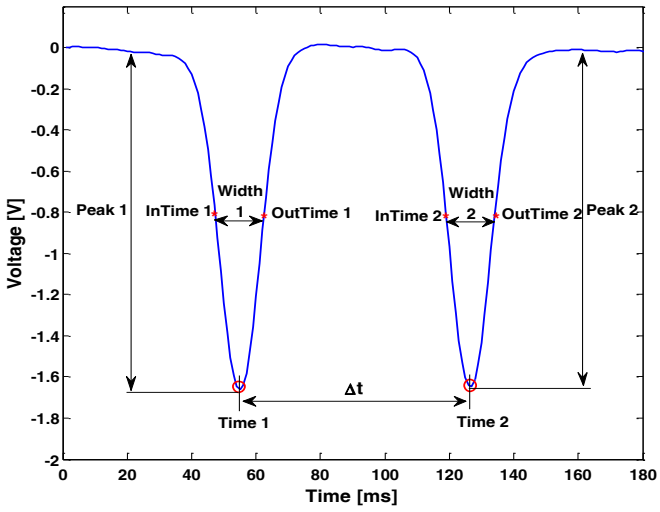


Figure 2.12: Features of detection signals from which information can be extracted for use by the event identifier.

The amplitudes of both the peaks are denoted as *Peak 1* [Volts] and *Peak 2* [Volts]; the times at which these peaks occur are *Time 1* and *Time 2* respectively. Every peak is associated with an in- time and out- time. The in and out- times are the times before and after the occurrence of every peak respectively. The in- times of the peaks are denoted as *InTime 1* and *InTime 2*. The out- times of the peaks are denoted as *OutTime 1* and *OutTime 2*. The time widths of peaks 1 and 2 are calculated at half the value of their maximum amplitudes and are denoted as *W1* and *W2*, respectively. The widths *W1* and *W2* are calculated from Equations 2.7a and 2.7b.

$$W1 = (Out\ Time\ 1 - In\ Time\ 1) \tag{2.7a}$$

$$W2 = (Out\ Time\ 2 - In\ Time\ 2) \tag{2.7b}$$

The normalized peak time difference Δt is calculated as

$$\Delta t = \frac{(Time\ 2 - Time\ 1)}{(Out\ Time\ 2 - In\ Time\ 1)} \tag{2.8}$$

The average velocity v_{avg} (μm/s) of a cell is calculated as

$$v_{avg} = \frac{Distance\ between\ the\ two\ detection\ sites}{(Out\ Time\ 2 - In\ Time\ 1)} \tag{2.9}$$

As shown in Figure 2.12, typical useful signals have: two peaks with the signal returning to zero in between, the signal crosses every voltage level (below zero) only four times. Signals shown in Figures 2.13 are rejected by the user for various reasons: multiple cells passing over the electrodes simultaneously, cells with a compromised membrane aggregating to form a lump, purges in the channel, and discontinuities due to noise from the electronic equipment or surroundings; these signals are rejected because they have multiple peaks and more than four crossings at any given voltage level. The features of acceptable and reject signals are explained in detail in Chapter 4.

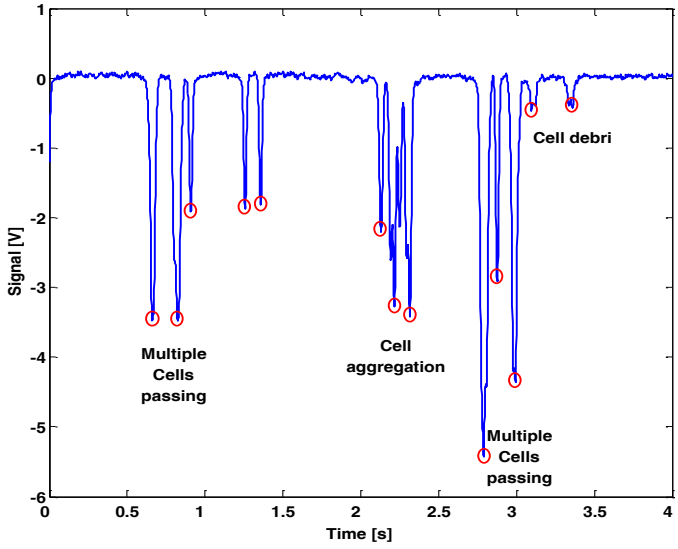


Figure 2.13: Signals rejected by the user during signal analysis.

Figures 2.14(a) and 2.15(a) depict the position of the cells in the channel corresponding to the detection signals shown in Figures 2.14(b) and 2.15(b) respectively.

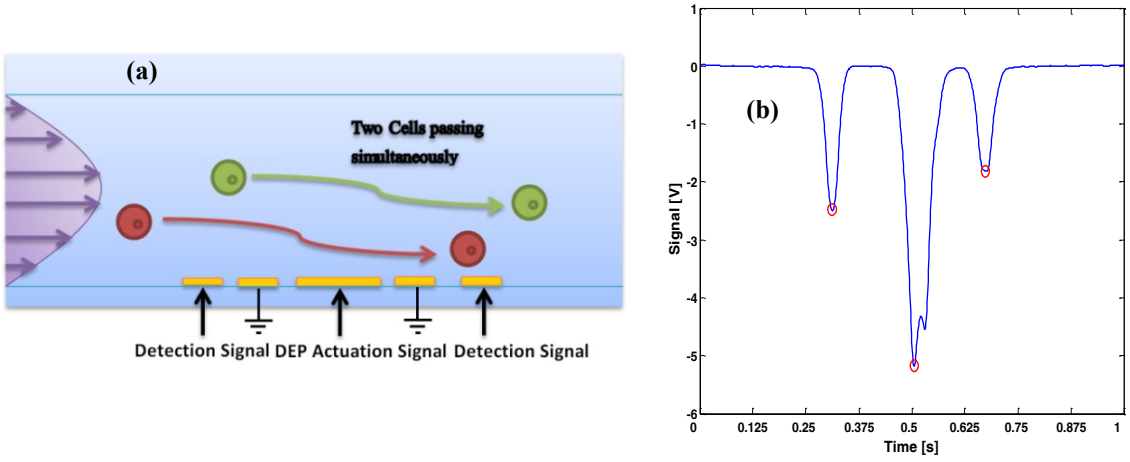


Figure 2.14: (a) Two cells passing simultaneously in the channel. (b) Detection Signal for two cells passing simultaneously.

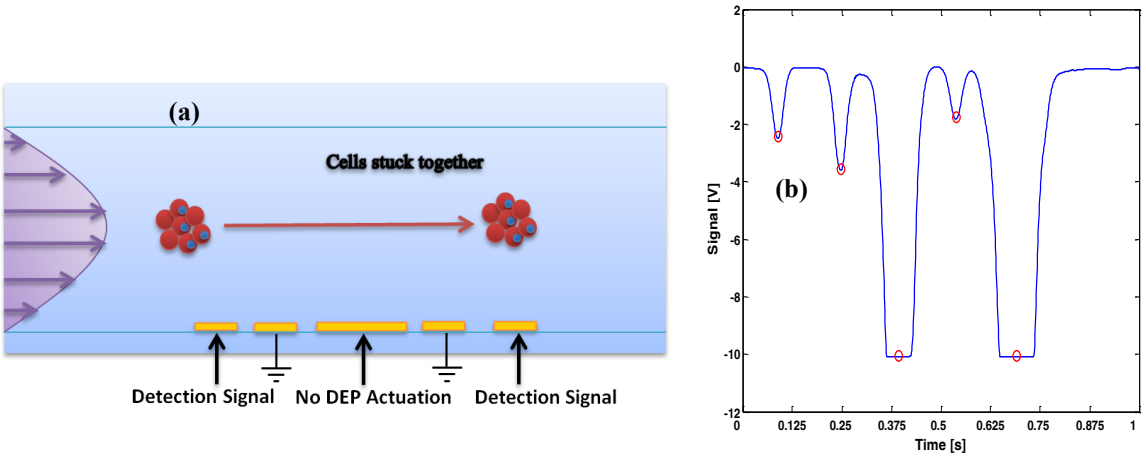


Figure 2.15: (a) An aggregate of cells flowing over the electrodes. (b) Detection Signal for an aggregate of cells.

The following chapter discusses an algorithm for the semi-automated identification of detection signals. The statistics obtained from the semi-automated data collection of

detection signals are compared to the statistics generated from the manual data collection of signals.

Chapter 3

Semi-Automated Event Identification of Detection Signals from a Dielectrophoretic Cytometer

An algorithm for semi-automated identification of detection signals obtained from a dielectrophoretic cytometer is presented in this chapter. Conventionally, detection signals taken from cells are reviewed and collected manually to extract signal peaks and times for further analysis. Manually identifying signals is a time consuming process and requires considerable human intervention. The goal of the semi-automated data collection method is to come up with an algorithm to identify a detection signal as an event, to reduce the signal analysis time and lessen human interference during signal post-processing and analysis. A detection signal contributed by one biological cell is defined as one event.

3.1 Data Acqusition

The microfluidic device, explained in Section 2.1 used along with an *RF* interferometer as described in [22] is designed to measure the dielectric properties of biological cells. The detection signals containing information about the dielectric properties are acquired in the time domain. The amplitudes of peaks, time between peaks, and peak widths are used to measure the force acting on the cell. Previous students have written programs to acquire detection signals from the *DEP* cytometer [31] [32]. The detection signals collected from the output of the *DEP* cytometer are acquired using a

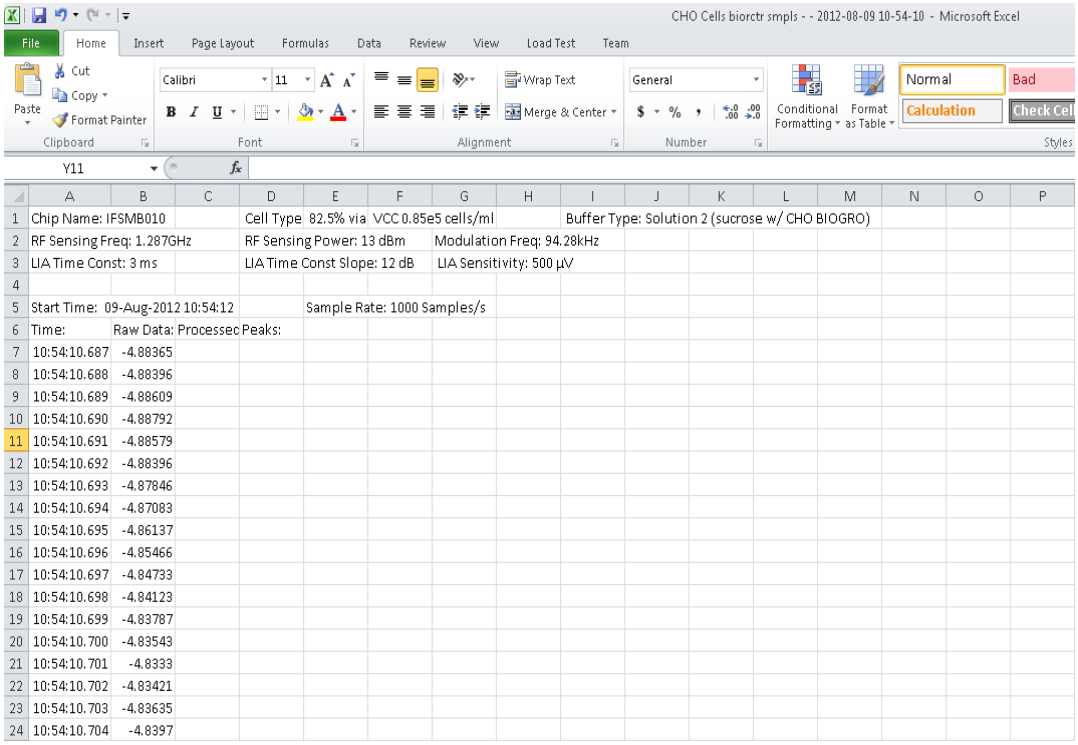
DAQ board which samples the real-time signals into digital samples for easier handling by a *PC*. The program collects the digital samples, marks them with time strings, stores them in small blocks, labels the data with the parameters of the experiment, and displays it during experimental data collection. The typical sampling rate is 1000 samples/sec. The program consecutively collects 30000 sample points and stores them in 30sec blocks of data so that it can be loaded into MATLAB and Microsoft Excel. The blocks of data are stored in .csv (comma separated variable) format and every 30s block constitutes one raw data file [31]. Figure 3.1, shows an example of the sampled real-time data file with details of the experimental parameters.

Another program was written to process real-time data post-experiment and to analyze it in order to study the dielectric properties of biological cells [31]. Real-time data often has some distortions due to purges in the channel, variation in the sensing frequency and *DEP* voltage, fluctuations from the pumping equipment and the environment in which the cytometer is operating in.

3.2 Manual Event Identification

Conventionally, the raw data containing the detection signals are post-processed by filtering the raw data, fitting the raw data to the base-line using a polynomial fit method, additionally filtering it with a moving average filter to smooth the data and to reduce noise, finding peaks and widths of the peaks. Once the events are identified after applying the post-processing algorithm, the user is required to manually select useful signals and eliminate signals caused due to irregularities during experimental data collection. The graphical user interface used by the manual event identification algorithm is shown in Figure 3.2. The raw data is displayed on the top axes and the post-processed data is

displayed on the bottom axes [31]. The peaks of the events identified by the peak determination algorithm [33] are marked by red circles (‘o’). The in and out-times of the events are marked by red asterisks (‘*’).



CHO Cells biorctr smpls - - 2012-08-09 10-54-10 - Microsoft Excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Chip Name: IFSMB010			Cell Type	82.5% via	VCC 0.85e5 cells/ml			Buffer Type: Solution 2 (sucrose w/ CHO BIOGRO)							
2	RF Sensing Freq: 1.287GHz			RF Sensing Power: 13 dBm			Modulation Freq: 94.28kHz									
3	LIA Time Const: 3 ms			LIA Time Const Slope: 12 dB			LIA Sensitivity: 500 µV									
4																
5	Start Time:	09-Aug-2012 10:54:12			Sample Rate: 1000 Samples/s											
6	Time:	Raw Data:	Processed	Peaks:												
7	10:54:10.687	-4.88365														
8	10:54:10.688	-4.88396														
9	10:54:10.689	-4.88609														
10	10:54:10.690	-4.88792														
11	10:54:10.691	-4.88579														
12	10:54:10.692	-4.88396														
13	10:54:10.693	-4.87846														
14	10:54:10.694	-4.87083														
15	10:54:10.695	-4.86137														
16	10:54:10.696	-4.85466														
17	10:54:10.697	-4.84733														
18	10:54:10.698	-4.84123														
19	10:54:10.699	-4.83787														
20	10:54:10.700	-4.83543														
21	10:54:10.701	-4.8333														
22	10:54:10.702	-4.83421														
23	10:54:10.703	-4.83635														
24	10:54:10.704	-4.8397														

Figure 3.1: Example of the sampled real-time data file with details of the experimental parameters.

The user selects a useful signal by clicking on the peaks, in-times and out-times of the signal marked with green circles and asterisks respectively. At the end of signal collection, peak amplitudes, peak times, in-times, and out-times of the analyzed data are saved in the .csv format. Manual data collection is a time consuming process so an alternative semi-automated data collection method was developed based on some aspects of the manual event identifier. The semi-automated data collection method has also reduced the data collection time by more than 50% (Table 3.3).

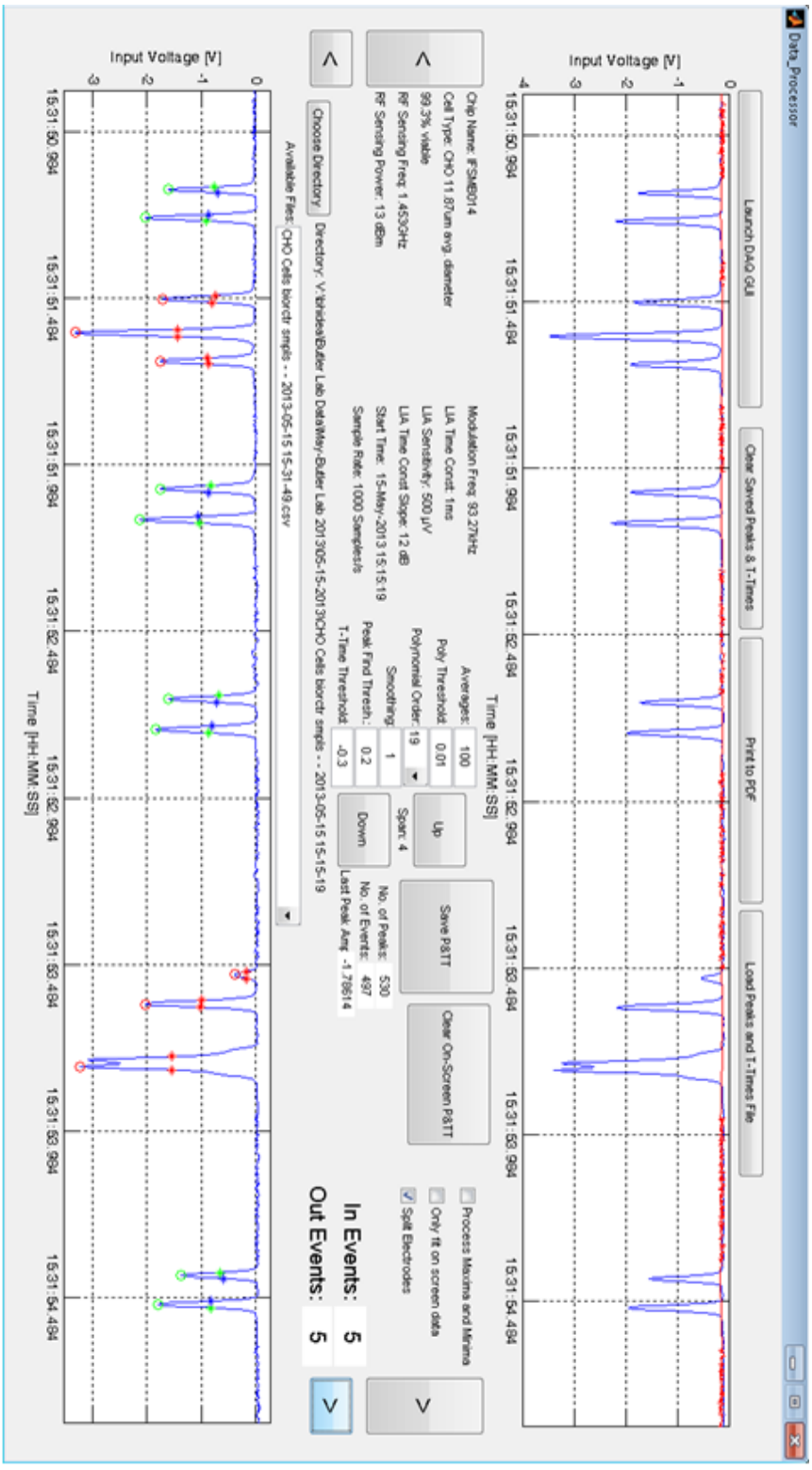


Figure 3.2: Post Processing using the Manual Event Identification program.

3.3 Semi-Automated Event Identification

3.3.1 Post-Processing Experimental Data

The data collection methods explained in this thesis make use of similar post-processing algorithms developed based on the post-processing algorithm used for manual event identification [31] but their analysis algorithms vary. The data collection methods are programmed to handle only one raw data file at a time. The raw data (Figure 3.3 (a)) is filtered by a moving average filter to get filtered data (Figure 3.3 (b)) which is subtracted from the raw data to get a base-line (Figure 3.3 (c)). Any points on base-line that exceed a certain threshold (0.1 in this case) are removed and then a polynomial is fit to the remaining base-line data points. The order of the polynomial is set to 3 after many trials. The polynomial fitted line is then subtracted from the raw data to move it up to a zero base-line. The base-line subtracted raw data is filtered by applying another moving average filter to smooth the raw data and to reduce noise. For extremely noisy raw data, the noise is further reduced by subtracting the mean value of the smoothed data from the smoothed data. Any data less than half the standard deviation value of the smoothed data is removed. An additional filtering is performed on the noise reduced smoothed data to identify events. Essentially, the noise reduced smoothed data is convolved with the rectangular window (Figure 3.3 (d)). The resulting window filtered output is the portion of the noise reduced smoothed data where it overlaps with the window and is zero-valued outside the interval [34]. However, there is a small time shift in the window filtered output which is corrected by subtracting the window size from the start time of the event.

A peak detection algorithm [33] is used to find peaks in the window filtered data. The time widths of these peaks are calculated at full width half maximum.

3.3.2 Event Identification Algorithm

An event identification algorithm, which identifies detection signals as events, is run on the window filtered data (Figure 3.3 (e)). This algorithm has two sub-functions: one finds the start points of events and another finds the end points of events. Different window functions are tested for event detection but filtered outputs for only two window functions are shown in Figures 3.4(a) and 3.4(b) for comparison. Rectangular window function is chosen because it pushes down the return to zero path between the two peaks which makes it easier for the algorithm to find only a unique start and end index for every event (Figure 3.4(a)). As shown in Figure 3.4(b), using a triangular window function does not push down the return to zero path in which case every event can have multiple start and end indices. Having more than one start and end index makes event identification complicated. The algorithm then finds the indices of the start and end points of events which are below a threshold set to -0.1. The start index of an event is the first point where it crosses the threshold and the end index is the last point below the threshold before it reaches back to baseline (Figure 3.3(d)). An event counter is set to ‘0’ when a 30s file is loaded into the memory; it increments by 1 every time a new event identified. Every event is associated with an event number. The start and the end index of an event are accessed by specifying its event number. An identified detection signal is the window filtered data from its start index to end index [34].

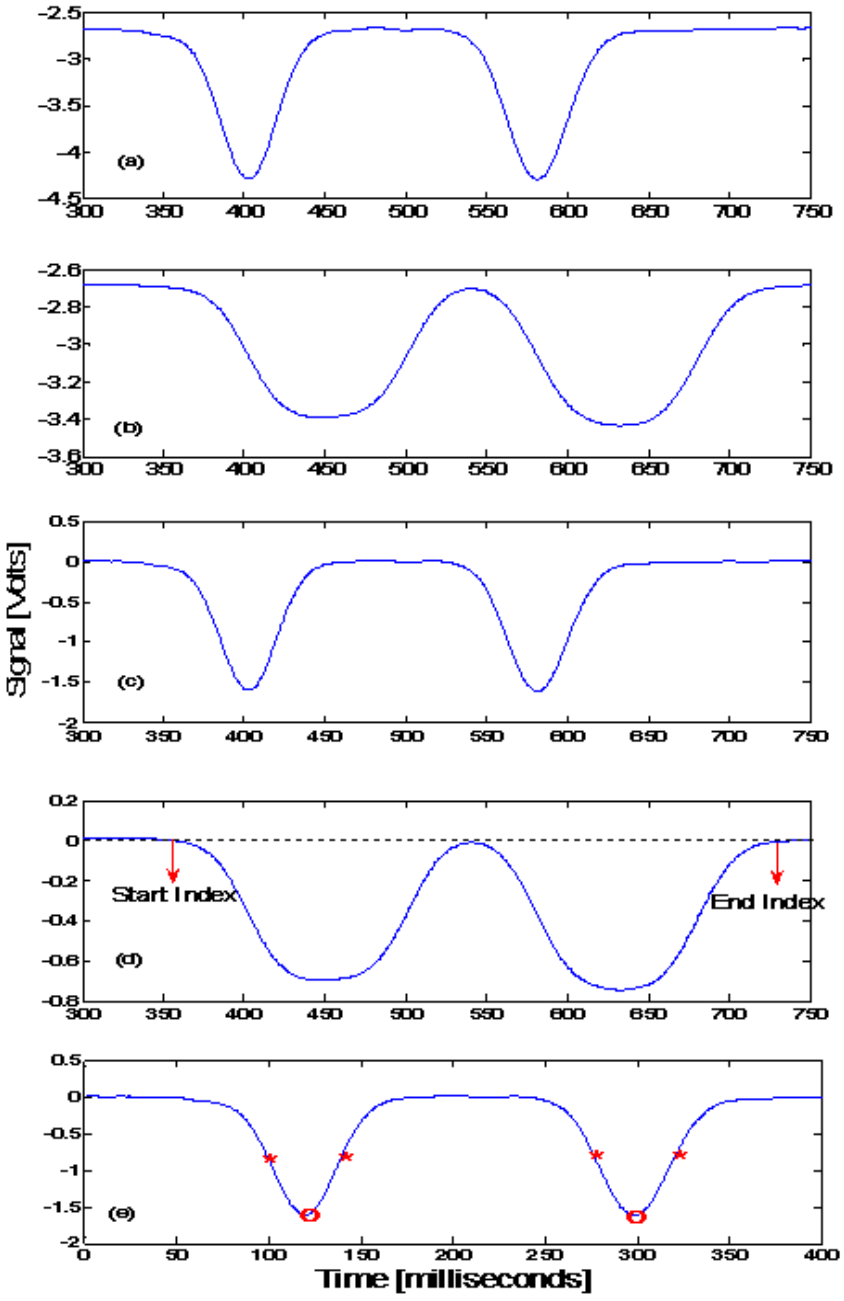


Figure 3.3: (a) Raw data (b) Data after filtering it with a moving average filter (c) Data fit to baseline (d) Data after filtering it with a rectangular window function of size 100ms (---- is the event detection threshold) (e) Detected event with peaks (‘o’) and in-out times (‘*’)

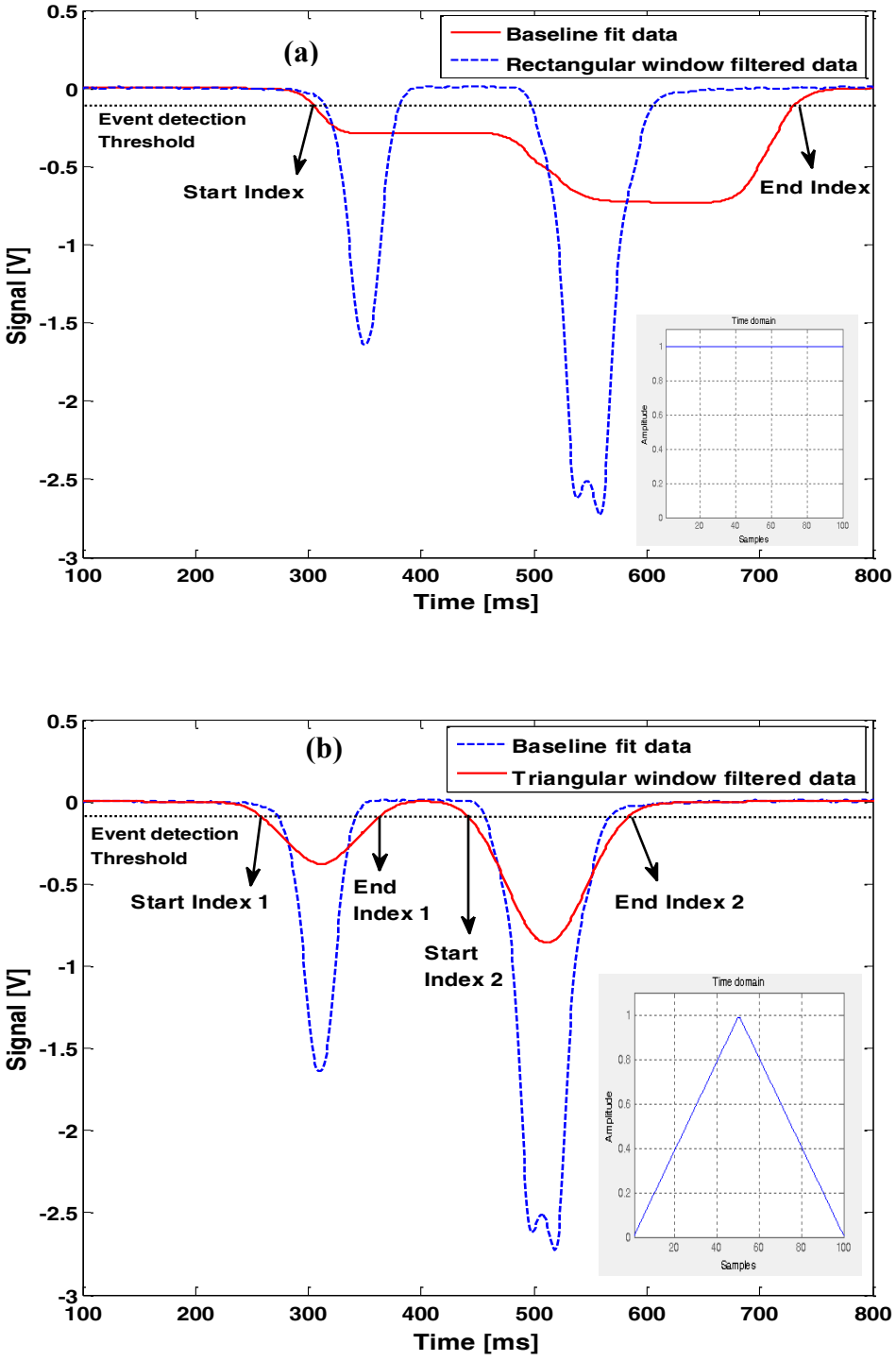


Figure 3.4: (a) Detection of events by rectangular window filtering. (b) Detection of events by triangular window filtering.

3.3.3 Signal Analysis

The flow diagram of how detection signals are identified as events and then analyzed using the semi-automated event identifier is shown in Figure 3.5.

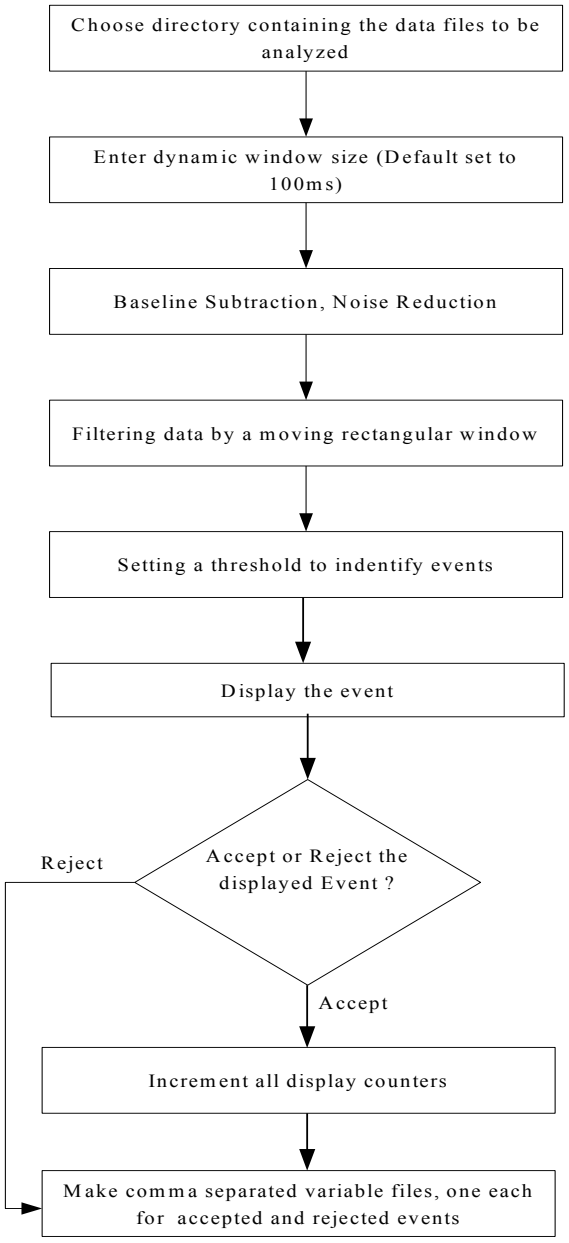


Figure 3.5: Flow diagram for the Semi-automated Event identification program.

On running the semi-automated event identification algorithm, a graphical user interface window opens up as shown in Figure 3.6.

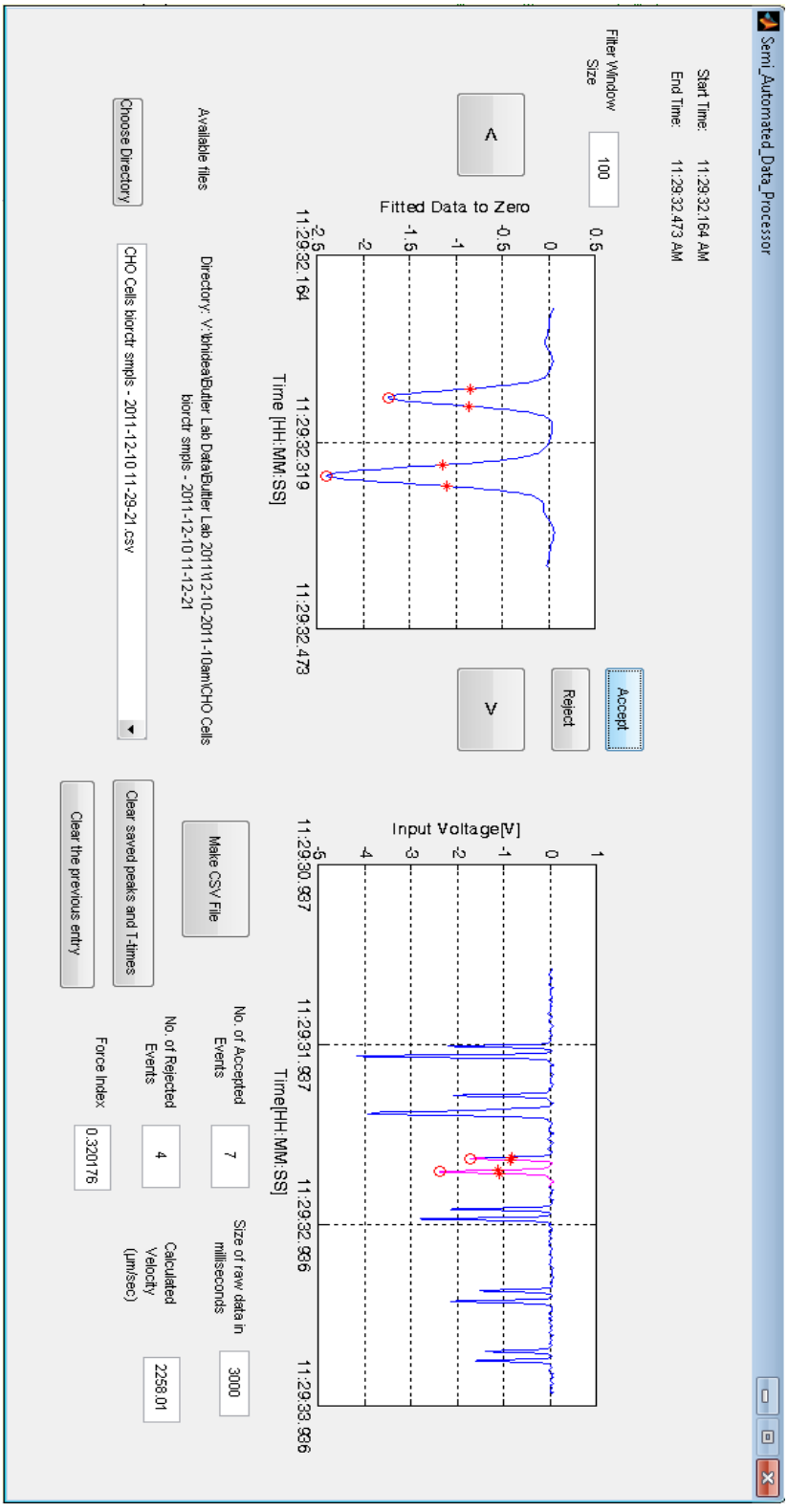


Figure 3.6: Post Processing using the Semi-Automated Event Identification program.

The experimental data to be analyzed is located by clicking the "Choose Directory" button and cruising through the directory to the proper location of the data. The data files with the correct format are loaded by scanning the folder. The drop down menu "Available Files" is then populated with the data files from the experiment. The user can enter their desired window size in the edit box associated with "Filter Window Size" text. The window size is the length of the rectangular window with which the data is filtered in the time domain. A fair assumption of the initial window size can be the time between the two peaks; the default value is set to 100ms. When a 30s file is loaded, the first identified event of that file is displayed on the left axes. The start ('Start Time') and end ('End Time') times of an event are displayed in the top right hand corner of the interface. The right axes on the interface displays filtered, base-line fit raw data for the next 3s span from the start time of the event displayed on the left axes. The time span can be varied by the user by specifying it in the 'Size of raw data in milliseconds' edit box. The current event being displayed on the left axes is highlighted in pink on the right axes. It is possible to navigate to the next or previous event by clicking on forward ('>') or backward ('<') scroll button. User input is required to accept useful signals and reject signals generated from multiple cells passing over the detection electrodes simultaneously, purges in the channel, and discontinuities due to noise. An event is accepted by clicking on the 'Accept' button and rejected by clicking on the 'Reject' button. The edit boxes 'No. of Accepted Events' and 'No. of Rejected Events' display the number of accepted and rejected events during data collection. The edit box 'Calculated Velocity' displays the velocity (μm/s), as calculated from Equation 2.3, of the cell whose

detection signal is displayed on the left axes. The “Make csv” button allows the user to save peaks, peak times, in- times and out- times of acceptable signals, as shown in Figure 3.7, in the .csv format. To begin a new collection, the ‘Clear saved peaks and T-times’ button is clicked to clear all saved peak amplitudes, peak times, in and out times. In case, if a reject signal is accidently accepted by the user, the ‘Clear the previous entry’ button clears all entries made by the reject signal from the accepted signals list [34].

	A	B	C	D	E	F	G	H	I	J	K	L
1	Chip Name: IFSMB010			Cell Type: Ch 100% viak VCC1.58e5 cells/ml					Buffer Type: solution 2 (sucrose w/ CHO BIOGRO)			
2	RF Sensing Freq: 1.2895GHz			RF Sensing Power: 13 dBm			Modulation Freq: 93.22kHz					
3	LIA Time Const: 3 ms			LIA Time Const Slope: 12 dB			LIA Sensitivity: 500 µV					
4	Peak Revision											
5	Start Time: 22-Feb-2013 11:26:07				Sample Rate: 1000 Samples/s							
6												
7	In Time 1	Peak 1	Time 1	Out Time 1	T_Times1	In Time 2	Peak 2	Time 2	Out Time 2	T-Times2	Force Index	Transition Time Ratio
8	11:30:09.306	-1.82598	11:30:09.314	11:30:09.323	0.017	11:30:09.362	-1.86448	11:30:09.370	11:30:09.379	0.017	0.02086372	1
9	11:30:09.908	-1.1815	11:30:09.917	11:30:09.925	0.017	11:30:09.973	-1.18668	11:30:09.982	11:30:09.991	0.018	0.004381478	1.058823529
10	11:30:10.281	-0.7847	11:30:10.289	11:30:10.298	0.017	11:30:10.349	-0.79208	11:30:10.357	11:30:10.367	0.018	0.009359775	1.058823529
11	11:30:11.185	-0.808	11:30:11.192	11:30:11.201	0.016	11:30:11.237	-0.81008	11:30:11.245	11:30:11.253	0.016	0.00257647	1
12	11:30:13.445	-0.47895	11:30:13.452	11:30:13.460	0.015	11:30:13.499	-0.46695	11:30:13.506	11:30:13.514	0.015	-0.025380665	1
13	11:30:17.798	-1.40631	11:30:17.807	11:30:17.816	0.018	11:30:17.867	-1.34067	11:30:17.876	11:30:17.885	0.018	-0.047790669	1
14	11:30:18.865	-0.90527	11:30:18.873	11:30:18.881	0.016	11:30:18.920	-0.90172	11:30:18.928	11:30:18.936	0.016	-0.003928773	1
15	11:30:19.492	-0.82905	11:30:19.502	11:30:19.511	0.019	11:30:19.567	-0.88421	11:30:19.576	11:30:19.586	0.019	0.064393791	1
16	11:30:21.891	-1.10575	11:30:21.901	11:30:21.910	0.019	11:30:21.959	-1.08358	11:30:21.968	11:30:21.978	0.019	-0.020253104	1
17	11:30:22.338	-2.00982	11:30:22.346	11:30:22.355	0.017	11:30:22.394	-1.93889	11:30:22.402	11:30:22.410	0.016	-0.035922558	0.941176471
18	11:30:24.882	-0.92753	11:30:24.890	11:30:24.898	0.016	11:30:24.942	-0.92427	11:30:24.950	11:30:24.959	0.017	-0.003515785	1.0625
19	11:30:25.447	-0.75988	11:30:25.456	11:30:25.466	0.019	11:30:25.523	-0.78967	11:30:25.532	11:30:25.543	0.02	0.038446549	1.052631579
20	11:30:28.968	-0.62049	11:30:28.977	11:30:28.985	0.017	11:30:29.031	-0.5498	11:30:29.039	11:30:29.047	0.016	-0.120818677	0.941176471
21	11:30:30.782	-1.40964	11:30:30.792	11:30:30.802	0.02	11:30:30.856	-1.27531	11:30:30.866	11:30:30.876	0.02	-0.100063749	1
22	11:30:36.932	-0.9473	11:30:37.038	11:30:36.987	0.055	11:30:37.029	-0.94221	11:30:37.107	11:30:37.048	0.019	-0.005384978	0.345454545
23	11:30:42.203	-0.93034	11:30:42.211	11:30:42.220	0.017	11:30:42.258	-0.98166	11:30:42.266	11:30:42.275	0.017	0.053681781	1
24	11:30:44.517	-0.94657	11:30:44.526	11:30:44.534	0.017	11:30:44.571	-0.94777	11:30:44.579	11:30:44.588	0.017	0.001268862	1
25	11:30:45.539	-0.81072	11:30:45.549	11:30:45.559	0.02	11:30:45.615	-0.92621	11:30:45.626	11:30:45.636	0.021	0.132982005	1.05
26	11:30:49.790	-1.11981	11:30:49.800	11:30:49.810	0.02	11:30:49.871	-1.12543	11:30:49.881	11:30:49.891	0.02	0.005001996	1
27	11:30:53.015	-1.10756	11:30:53.022	11:30:53.030	0.015	11:30:53.068	-1.21393	11:30:53.075	11:30:53.084	0.016	0.091635662	1.066666667
28	11:30:54.982	-0.73268	11:30:54.989	11:30:54.997	0.015	11:30:55.032	-0.76627	11:30:55.039	11:30:55.047	0.015	0.044818412	1
29	11:30:56.146	-1.47742	11:30:56.154	11:30:56.162	0.016	11:30:56.201	-1.45309	11:30:56.209	11:30:56.217	0.016	-0.016606723	1.066666667
30	11:30:56.760	-0.6398	11:30:56.767	11:30:56.775	0.015	11:30:56.813	-0.62198	11:30:56.821	11:30:56.829	0.016	-0.02825128	1.066666667
31	11:30:57.743	-0.59042	11:30:57.750	11:30:57.757	0.014	11:30:57.795	-0.61667	11:30:57.802	11:30:57.810	0.015	0.043485523	1.071428571
32	11:31:02.014	-1.41248	11:31:02.022	11:31:02.032	0.018	11:31:02.076	-1.37315	11:31:02.084	11:31:02.093	0.017	-0.028241068	0.944444444
33	11:31:02.922	-0.69998	11:31:02.929	11:31:02.938	0.016	11:31:02.970	-0.68985	11:31:02.978	11:31:02.986	0.016	-0.014580005	1
34	11:31:03.448	-0.99442	11:31:03.456	11:31:03.465	0.017	11:31:03.511	-0.97621	11:31:03.519	11:31:03.528	0.017	-0.018480455	1
35	11:31:06.445	-0.85609	11:31:06.452	11:31:06.460	0.015	11:31:06.495	-0.84149	11:31:06.502	11:31:06.510	0.015	-0.01721029	1
36	11:31:08.741	-0.49299	11:31:08.748	11:31:08.756	0.015	11:31:08.791	-0.45779	11:31:08.797	11:31:08.805	0.014	-0.074039056	0.933333333
37	11:31:09.168	-1.19837	11:31:09.176	11:31:09.184	0.016	11:31:09.219	-1.17761	11:31:09.226	11:31:09.235	0.016	-0.017472374	1
38	11:31:11.277	-1.44721	11:31:11.285	11:31:11.293	0.016	11:31:11.332	-1.44178	11:31:11.340	11:31:11.348	0.016	-0.003765334	1
39	11:31:11.471	-0.88393	11:31:11.479	11:31:11.487	0.016	11:31:11.530	-0.87426	11:31:11.538	11:31:11.547	0.017	-0.010993196	1.0625

Figure 3.7: CSV file containing the processed data collected using the Semi-Automated Event Identification program.

3.4 Results and Conclusions

To compare the semi-automated and manual event identification methods, signals collected from an experiment on CHO cells are used. These experiments are used to estimate the percentage of viable cells. Detection signals are experimentally collected for unactuated cells and cells actuated by *DEP* forces. These signals are classified manually and also by the semi-automated technique described in this chapter. In order to verify that the force index distributions of *DEP* and Control group signals are not skewed by semi-automation, normalized force index distributions for both manual and semi-automated collections are plotted as shown in Figures 3.8 and 3.9. Normalization involves scaling the number of events in each force index bin by the total of number of events in all force index bins. Both the distributions are almost similar. The semi-automated identifier misses out on identifying some events for reasons: (1) they have lower amplitude than the set threshold; (2) they lie on discontinuities; (3) only half an event is identified as a result of a small window size; (4) more than one event is identified as a result of bigger window size.

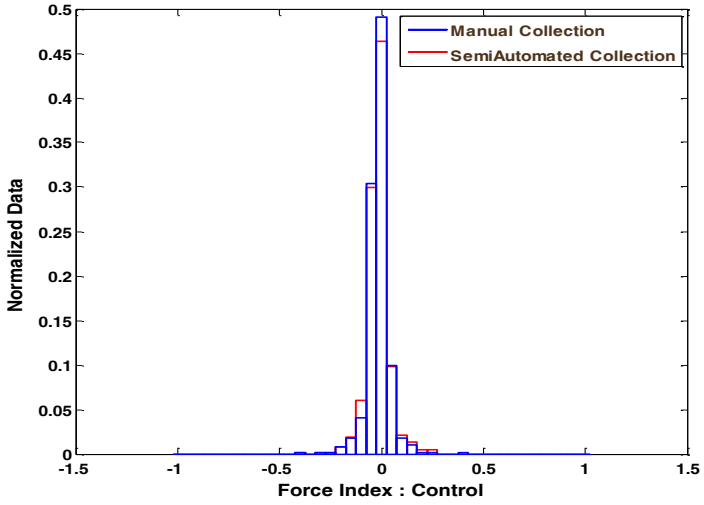


Figure 3.8: Comparison of control group force index distributions for manual and semi-automated data collections.

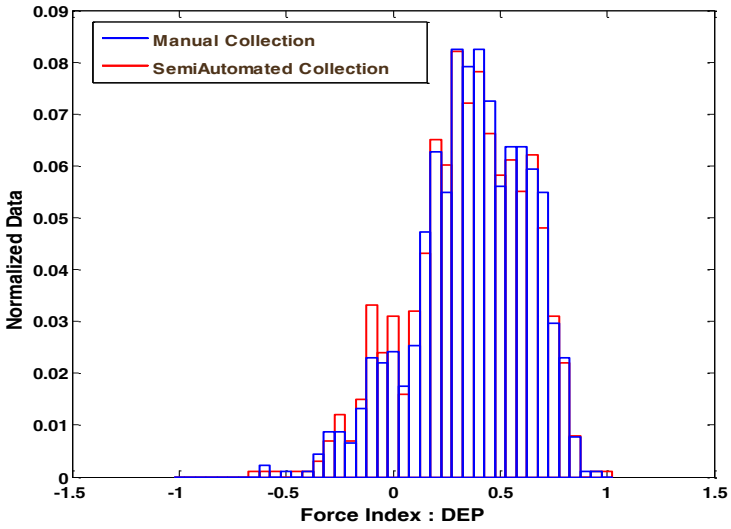


Figure 3.9: Comparison of *DEP* force index distributions for manual and semi-automated data collections.

To validate the distribution obtained from the semi-automated data collection, the uncertainty in the normalized number of events in each bin is calculated by Equation 3.1.

$$Uncertainty = \frac{N \pm \sqrt{N}}{Total\ number\ of\ events} \tag{3.1}$$

N denotes the number of events in each force index bin. This calculation is done for force index bins with more than 10 events for an accurate comparison. These uncertainties are represented with error bars as shown in Figure 3.10. The uncertainties in percentage viabilities for both semi-automated and manual collection methods are ±3%.

A two-sample *T*-test performed separately on control group and *DEP* data collected from both the manual and semi-automated methods shows that null hypothesis is supported. Here, the null hypothesis is that the data collected from both the manual and

semi-automated collections are the same. In a *T*-Test, the null hypothesis is tested by determining the likelihood that a test statistic could have been selected, if the null hypothesis regarding the data set were to be true [35]. The null hypothesis is a parameter of the data set that is assumed to be true. Assuming that the two distributions have the same variance, the *t* statistic can be calculated as

$$t = \frac{\bar{X}_{dataset1} - \bar{X}_{dataset2}}{A.B} \tag{3.2}$$

$$A = \sqrt{\frac{(n_1-1)s^2_{dataset1} + (n_2-1)s^2_{dataset2}}{(n_1+n_2-2)}} \tag{3.3}$$

$$B = \sqrt{\frac{1}{n_1} + \frac{1}{n_2}} \tag{3.4}$$

$\bar{X}_{dataset1}$ and $\bar{X}_{dataset2}$ are the mean values of the datasets obtained from manual and semi-automated data collections respectively; $s^2_{dataset1}$ and $s^2_{dataset2}$ are variances of the datasets obtained from manual and semi-automated data collections respectively; n_1 and n_2 are the number of samples in datasets 1 and 2. The degree of freedom *df* is calculated as expressed in Equation 3.5

$$df = (n_1 + n_2 - 2) \tag{3.5}$$

The probability of obtaining a test statistic, given that the null hypothesis is true, is stated by the *p*-value [35]. The *p*-value is a probability and it varies between 0 and 1. The *p*-value is calculated from the *t* statistic and the degree of freedom. If the *p*-value is less than or equal to 5% ($p <= .05$), the null hypothesis is rejected. If the value is greater than 5% ($p > .05$), the null hypothesis is retained. Significance level refers to a criterion depending on which a decision is made regarding the null hypothesis and its typical value

is 5%. The null hypothesis is rejected when the significance level is reached; it is accepted when the significance level cannot be reached [35].

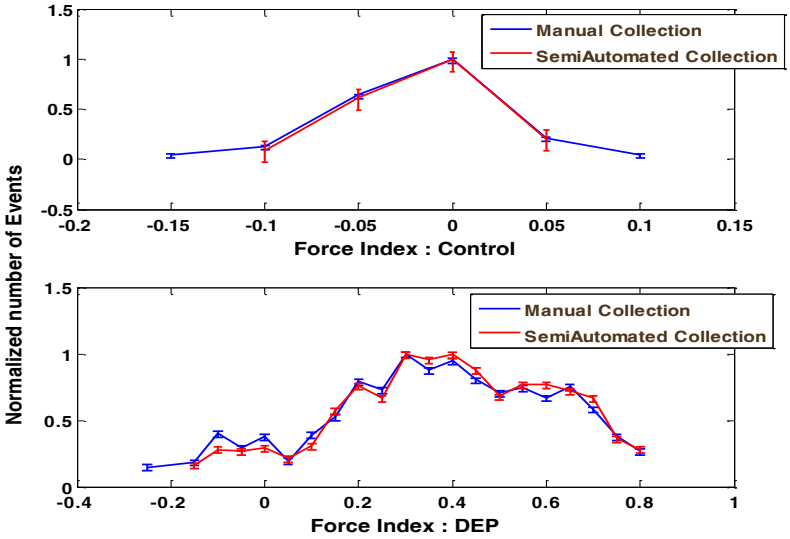


Figure 3.10: Uncertainty comparison in number of Control group and *DEP* events for semi-automated and automated collections.

Table 3.1: Comparison of Control group statistics for manual and semi-automated data collections.

Collection Method	**Statistics : Control Group**			***T*-test**	
	N	**μ_Φ**	**σ_Φ**	***H***	***p*- value**
Manual	578	-0.016	0.05	0	0.784
Semi-Automated	526	-0.015	0.05		

Table 3.2: Comparison of *DEP* statistics for manual and semi-automated data collections.

Collection Method	**Statistics: *DEP***				***T*-test**	
	N	**μ_Φ**	**σ_Φ**	**%Viability**	***H***	***p*-value**
Manual	998	0.36	0.26	89.8±3%	0	0.217
Semi-Automated	852	0.35	0.26	88.8±3%		

The semi-automated event identification algorithm is capable of identifying 87.8% of the events that are identified by the manual method. In addition, mean values μ_Φ and standard deviations σ_Φ of the force index distributions are calculated along with their percentage viabilities and compared for both the manual and semi-automated collections in Tables 3.1 and 3.2. The hypothesis test results H=0 and p-values indicate that null hypothesis cannot be rejected.

The time required by the semi-automated identifier for collecting detection signals is three times less when compared to the time taken by the manual event identifier to analyze the same data. The compromise in efficiency is acceptable because the time required for signal collection is reduced significantly. The time required for collecting 100 detection signals by both the manual and semi-automated methods is tabulated in Table 3.3.

Table 3.3: Timed Collection of 100 detection signals by both manual and semi-automated data collection methods.

Data Collection Method	Number of Signals	Collection Time (seconds)
Manual	100	476
Semi-Automated	100	162

The next chapter discusses an automated method based on pattern recognition for classifying acceptable and reject-able detection signals into their respective categories. The steps involved in the design of the automated signal classifier are explained in detail. The analysis results obtained from the automated classification of detection signals are compared to the results obtained from the manual and semi-automated collections of the same detection signals.

Chapter 4

An Artificial Neural Network-based Signal Classifier for Automated Identification of Detection Signals from a Dielectrophoretic Cytometer

A neural network-based automated detection signal classifier is presented in this chapter. The ANN-based classifier recognizes useful detection signals based on their patterns. The requirements of the ANN-based detection signal classifier are: smaller loss percentage in signal misclassification as a result of automation, automation should not skew the force index distributions when compared to the distributions generated from the manual and semi-automated data collection methods, the signal classifier must be robust i.e. the algorithm for signal classification must be effective when it is tested on a new independent data set, operating this classifier must need minimum human intervention, and analyzing large volumes of data should require less time when compared to other analysis techniques explained in Chapters 3.

4.1 Comparing Biological and Artificial Neurons

The basic component of an ANN is an artificial neuron, which is a mathematical interpretation of a biological neuron. In a biological neuron, as illustrated in Figure 4.1

[36], the neuron cell body houses the nucleus which functions as a summing junction. The dendrites act as inputs to the ANN. The signal to an axon is the net input. A structure called the synapse, present at the terminal end, passes electrical information from the axon of one neuron to either activate or inhibit the neighboring neuron. If the potential on the neuron cell body is above some threshold potential, then the electrical impulses travel down the axon of one neuron to activate the neighboring neuron via its terminal ends. Similarly, if the output signal of one artificial neuron is above a threshold value then it is passed on as an input to the adjacent AN. Human learning occurs by strengthening the synaptic connections between two active neurons. In ANN's, weights are equivalent to the electrical information [37].

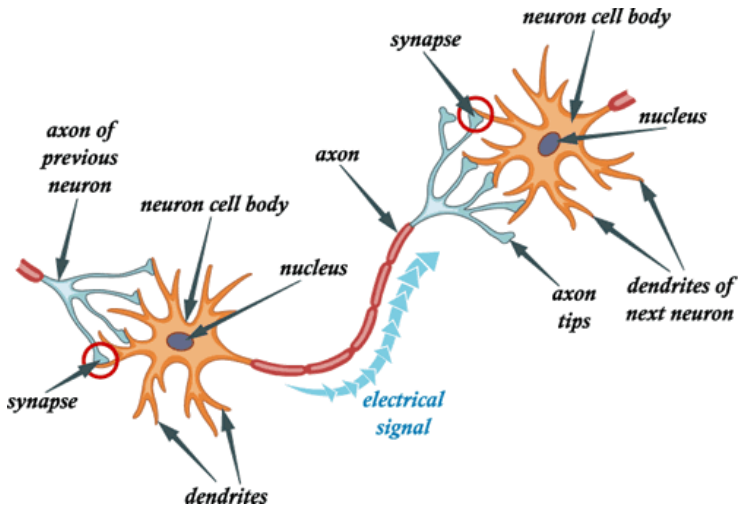


Figure 4.1: Communication between two biological neurons by transmission of electrical information from one active neuron to another [36].

4.2 Artificial Neuron

An artificial neuron is modelled as shown in Figure 4.2 [38]. The three important operations that occur in an AN '*j*' are: the inputs ($x_1, x_2, x_3....x_n$) and the weights

(w_{j1},w_{j2},w_{j3},....w_{jn}) are sent to the summing junction where they are dot multiplied, the weighted inputs are added to the bias value to form the net input 'v_j'and, the net input is passed through a transfer function which then produces the output 'y_j'.

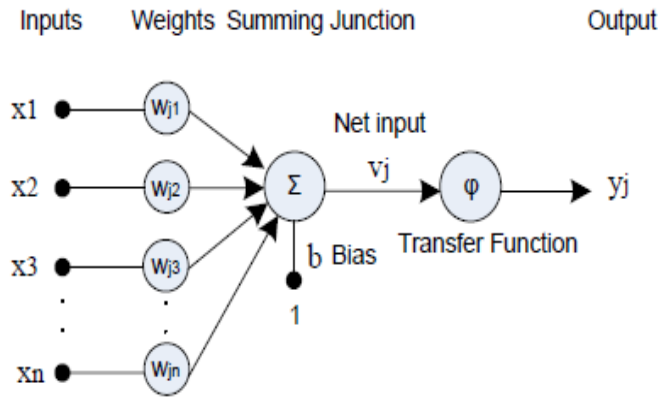


Figure 4.2: Model of an artificial neuron [38]. The neuron has 'x_n' inputs, 'w_{jn}' weights, bias 'b', net input 'v_j', transfer function φ, and an output 'y_j'.

The expression for the output y_j is

$$y_j = \varphi\left(\sum_{k=1}^{n}\left(w_{jk}x_k\right) + b\right) \tag{4.1}$$

A generalized form of Equation 4.1 is expressed as

$$y = \varphi(wp + b) \tag{4.2}$$

By adjusting the weights and the biases, the network is trained to show a particular behaviour neccesary for performing a specific task. The transfer function maps the net input to a value within its range. Some examples of transfer functions are: linear function, step function, log-sigmoid function, and symmetric sigmoid function. Every neuron layer includes weights multiplied by the inputs, biases, summing functions, net inputs, transfer functions, and outputs. The outputs of one layer are passed on as inputs to the subsequent

layer. Figure 4.3 shows a neuron layer and its simplified equivalent [38]. Weights amplify or dampen the inputs, and the bias value shifts the transfer function curve to the left or right of zero which is important for successful learning. The input x to a neuron layer is a vector consisting of n elements. The weight matrix w has dimensions $m \times n$ where m is the number of neurons in that particular layer and n is the length of the input vector. The number of inputs n and the number of neurons m are not necessarily equal. The scripts p and q attached to the weight matrix W indicate the source and destination of the weights respectively. The bias value b multiplied by a constant input 1 is given to every neuron in the layer. The vector input is multiplied by a weight matrix w, then summed with an m- length bias vector to generate an m- length net input vector. The m-length net input vector is passed through a transfer function φ to generate an m- length output y vector as expressed in Equation 4.1.

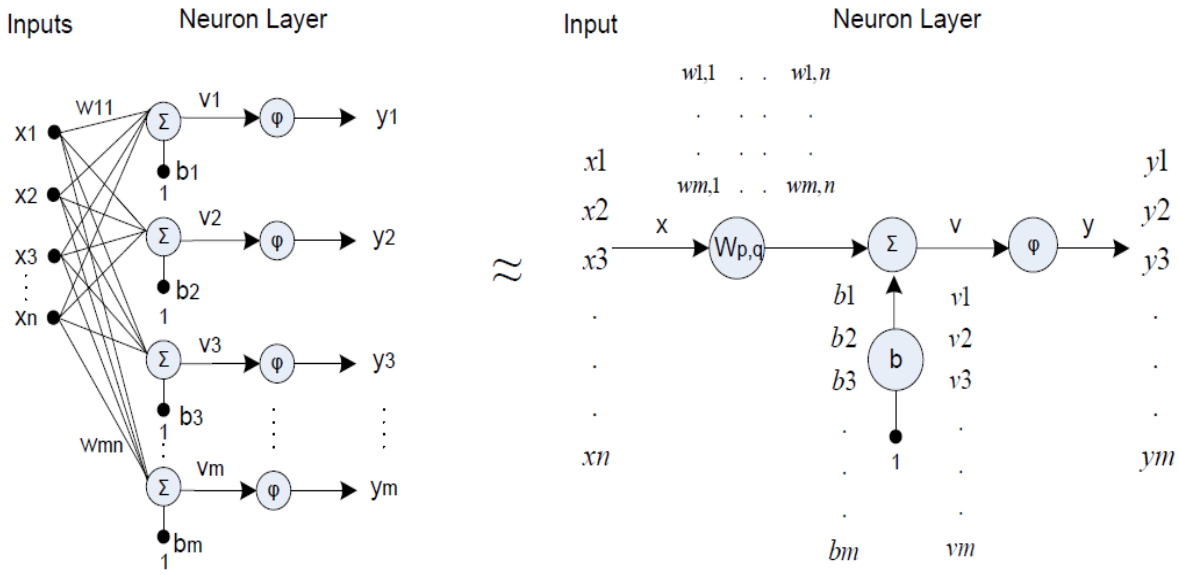


Figure 4.3: One layer of neurons and its simplified equivalent with dimensions of all parameters [38].

4.3 Designing an Artificial Neural Network

Artificial neural networks are designed to mathematically replicate the functioning of human brain. ANN's are capable of extracting complex patterns and features efficiently with minimal computational power. ANN's are able to accurately classify detection signals using historical data to train the classifier. The basic steps involved in the design of the artificial neural network are: pre-processing to select the appropriate features of detection signals to be presented to the ANN as training data, building the network, configuring the network, initializing the network parameters i.e. weights and biases, training the network, testing the network, and then using the network on a new dataset to classify detection signals into two target categories, viz., accept signals and reject signals [38].

The initial step in designing an ANN is to collect training data. The training data consists of training inputs and targets. Prior to training, the training data is divided into three subsets: Training, Validation and Test. The training set is used to train the network by adjusting the weights. The validation set is used to minimize data overfitting by the network and to determine the performance of the network. The test set, not included in the training, tests the prediction power of the network. The training inputs are normalized before they are applied to the network in order to prevent the transfer function from saturating. Training targets are also normalized so that the network output falls in the normalized range. The network outputs are de-normalized before using the network to train other data sets. The training inputs and targets have to be organized appropriately as input and target matrices before being presented to the network [38].

Building, Configuring and initializing a network involves deciding: the type of neural network, number of neuron layers, number of neurons in each layer, connections between the neuron layers, dimension of the weights and bias matrices, dimensions of the network's input and output, setting the initial values of network's adjustable parameters viz. weights and biases, the weight multiplication function, individual weight/bias updating algorithm (learning algorithm), the transfer function, type of performance function, training method and training algorithm.

Training the network is a process of adjusting the network parameters (weights and biases) to make the network learn, memorize and form a relationship between training inputs and targets. Training inputs are the features of detection signals and training targets are scalar values that represent the categories ('1' for accept and '0' for reject) into which these training inputs are classified. Based on the training inputs and targets, the ANN will classify other detection signals as accepted or rejected. A network optimization method adjusts the network's weights and biases until the performance of the network is optimal. The network is then simulated to compute its response to other unseen data [38].

4.4 Backpropogation Algorithm using Gradient descent optimization method

Multilayered feed-forward ANN's used extensively for classifications based on pattern recognition are chosen to classify detection signals because they are static in nature and less complicated. In feed-forward networks, the information flows only in the forward direction from the input nodes to the output layer via the hidden layers. The hidden layers

of the feed-forward network have non-linear transfer functions to introduce non-linearity into the ANN so that it learns the non-linear relationships between network inputs and outputs. A training algorithm trains the network to categorize detection signals based on a relationship that best maps training inputs to its outputs. Training algorithms are mainly divided into: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning is a very common method for training ANN's for classification based on pattern recognition. In supervised learning, as shown in Figure 4.4, the training data is first presented to the network for training. The difference, which is also the error, between the network output and the targets is computed and used by the training algorithm to adjust the weights. The MSE function calculates the average squared error, for n training samples, between the network outputs y and the training targets t as expressed in Equation 4.3.

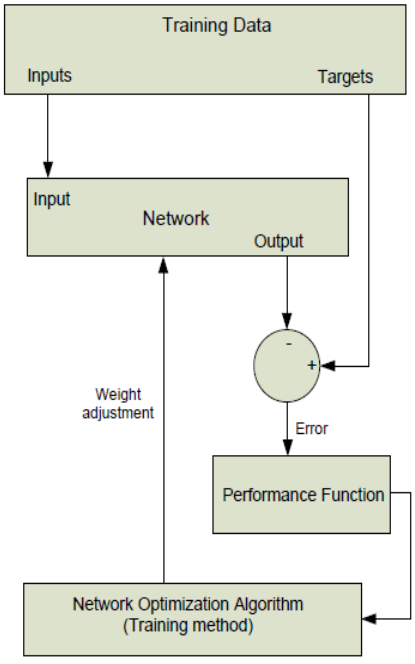


Figure 4.4: Steps involved in training ANN's by supervised learning.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(t_i - y_i)^2 \tag{4.3}$$

The feed-forward ANN is trained in the batch training mode where all inputs are applied to the network before the weights are updated. Batch training is a faster method of training and results in very small errors [38] [39]. A standard network optimatization algorithm viz. gradient descent backpropogation algorithm is used to optimize the network's performance and is described below.

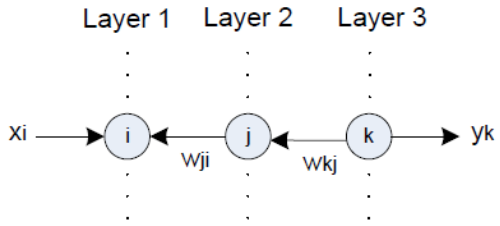


Figure 4.5: A three-layered feed-forward network with a neuron *i* in layer 1, a neuron *j* in layer 2 , and a neuron *k* in layer 3.

The flow of the backpropogation algorithm is: feed-forward propogation to generate ouputs by applying input patterns and initial weights, backward propogation of errors from layer 3 to layer 2, backward propogation of errors from layer 2 to layer 1, and then updating weights and biases [40].

Consider a three-layered feed-forward network with neurons *i*, *j* and *k* in layers 1, 2 and 3 respectively as shown in Figure 4.5. The input to neuron *i* in layer *1* is x_i and y_k is the output from neuron *k* in layer 3. The inputs to the hidden layers 2 and 3 are z_i and z_j respectively. The MSE function *E* is expressed in Equation 4.4. From Equations 4.1 and 4.2 , the net input v_j and the output z_j for layer 2 are expressed as

$$E = \frac{1}{p}\sum_{k=1}^{p}(t_k - y_k)^2 \tag{4.4}$$

$$v_j = \sum_i (w_{ji} z_i) \tag{4.5}$$

$$z_j = \varphi(v_j) \tag{4.6}$$

Equations 4.5 and 4.6 are obtained from the feed-forward computation.

The gradient which is the derivative of E with respect to w_{ji} according to the chain rule of partial derivatives is as expressed in Equation 4.7.

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial v_j}\frac{\partial v_j}{\partial w_{ji}} = \delta_j z_i \tag{4.7}$$

where $\frac{\partial E}{\partial v_j} = \delta_j$ and from differentiating Equation 4.4 we have $\frac{\partial v_j}{\partial w_{ji}} = z_i$.

The initial weights and biases are a matrix of random values. The output y_k from layer 3 is expressed as

$$y_k = \varphi(v_k) \tag{4.8}$$

The transfer function φ is a differentiable function. For neuron k in layer 3, δ_k is calculated as

$$\delta_k = \frac{\partial E}{\partial v_k} = \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial v_k} = \frac{\partial E}{\partial y_k}\varphi'(v_k) \tag{4.9}$$

Equation 4.9 is the backward propogation of errors from layer 3 to layer 2.

The net input v_k of neuron k is layer 3 is expressed as follows

$$v_k = \sum w_{kj}\ \varphi(v_j) \tag{4.10}$$

For neuron j in the hidden layer 2, δ_j is calculated as expressed below

$$\delta_j = \frac{\partial E}{\partial v_j} = \sum_k \frac{\partial E}{\partial v_k}\frac{\partial v_k}{\partial v_j} = \sum_k \delta_k \frac{\partial v_k}{\partial v_j} \tag{4.11a}$$

$$\delta_j = \varphi'(v_j)\sum_k w_{kj}\delta_k \tag{4.11b}$$

Equations 4.11a and 4.11b represent the backward propogation of errors from layer 2 to layer 1. Similarly, δs for all the neurons of the hidden layers can be calculated by propogating error in backward direction from the layers succeeding it. After the derivatives of the error function with respect to the weights are obtained, the weights are updated to decrease the error. The gradient descent method is used to compute the gradients and update the network weights and biases [40]. In the gradient descent algorithm, weights and biases are updated in the negative direction since the error decreases rapidly in that direction. For a neuron j in layer 2, the updated weight $\Delta w_{ji}^{updated}$ is expressed as:

$$\Delta w_{ji}^{updated} = -\eta \frac{\partial E}{\partial w_{ji}} * + \alpha * \Delta w_{ji}^{previous} \tag{4.12}$$

where η is the learning rate of the network and also the step size for the next iteration, α is the momentum constant to stabilize the network incase the learning rate is very high, $\Delta w_{ji}^{previous}$ is the change in weight from the previous iteration, and $\partial E/\partial w_{ji}$ is the gradient of error with respect to the weight as calculated from Equation 4.7. The default values set for learning rate and momentum constant are 0.01 and 0.9 respectively. The magnitude of the gradient is monitored to halt the training process. With every iteration (or epoch) of the algorithm, the gradient gets smaller as the performance begins to reach a minimum. The gradient descent method used to update weights in the standard back-propagation algorithm does not guarantee convergence to global minima and is very slow in reaching convergence. Therefore, a better method known as the scaled conjugate gradient method is used to update the weights and is explained briefly in the next section [38] [40].

4.4.1 Scaled Conjugate Gradient Optimization Method

Scaled Conjugate Gradient backpropogation algorithm is a more complex form of gradient descent backpropogation algorithm [41] and is used for training large datasets because they are fast and require less memory. The SCG algorithm builds a set of weight vectors in a weight space which constitute a conjugate system. The SCG algorithm can be briefly described as an iterative optimization method in which the error function is minimized with respect to a current point in the weight space. The next current point is determined by finding a search direction and then a step size. The search direction is determined by looking for a direction in the weight space until the gradient of error is perpendicular to that direction. The step size is then determined by deciding how far to go in this search direction. Unlike gradient descent algorithm, the SCG speeds up training by first finding a direction and then scaling the step size rather than finding a gradient in every iteration [41].

4.5 Architecture of the ANN for Automated Signal Classification

The number of neuron layers and the number of neurons per layer are important parameters to be determined in the design of the network; these parameters are dependent on the application and the number of training samples. For the purpose of classifying detection signals based on their patterns, one hidden layer of neurons is sufficient. Using several hidden layers slows down the training process and the training algorithm requires many time-consuming iterations to converge [38].

The number of neurons in the input and output layers are determined by the sizes of the training inputs and targets respectively. Using fewer neurons for a dataset results in an under trained network because the number of neurons are not sufficient to properly identify the detection signals. Using too many neurons over trains the network resulting in two problems: 1) the information in the training data becomes inadequate to train all the neurons of the hidden layers; 2) the neurons increase the time required to train the network [38]. The best method to choose the number of neurons is by monitoring the performance of the network and the percentage of training data incorrectly classified by the network. Prior to setting the number of neurons in the hidden layer, suitable training data features have to be selected as explained in Section 4.6. The most commonly used performance function MSE is used to evaluate the performance of the feed-forward network. As shown in Figure 4.6, 20 neurons are chosen for the hidden layer because at this number both the performance and percentage of misclassified training data are low. The network becomes unstable and the error on the network oscillates when more than 20 neurons are used.

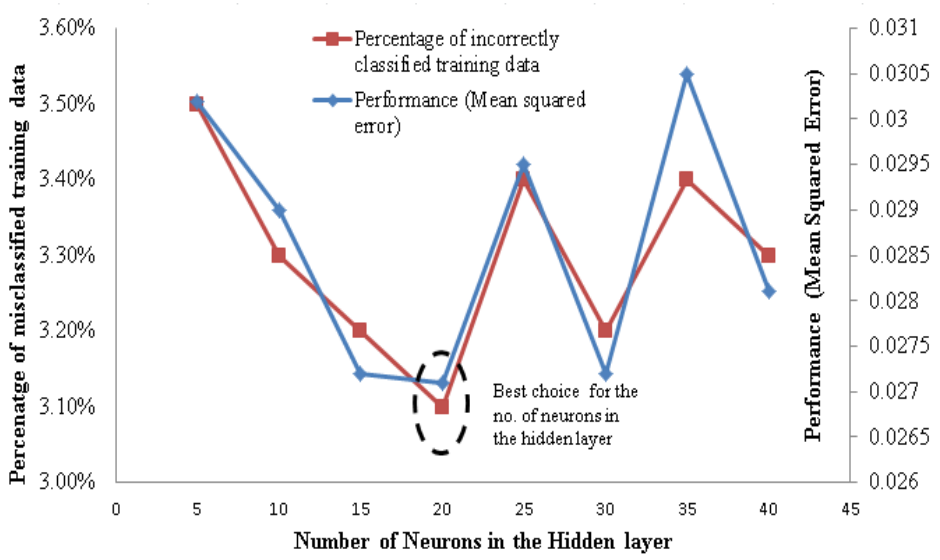


Figure 4.6: Choice for the number of neurons in the hidden layer is 20 where both the performance and percentage of misclassified training data are low.

The network designed for automated classification of detection signals, as shown in Figure 4.7, is a two-layered feed-forward ANN consisting of one hidden layer and one output layer; the architecture of this network is shown in Figure 4.8.

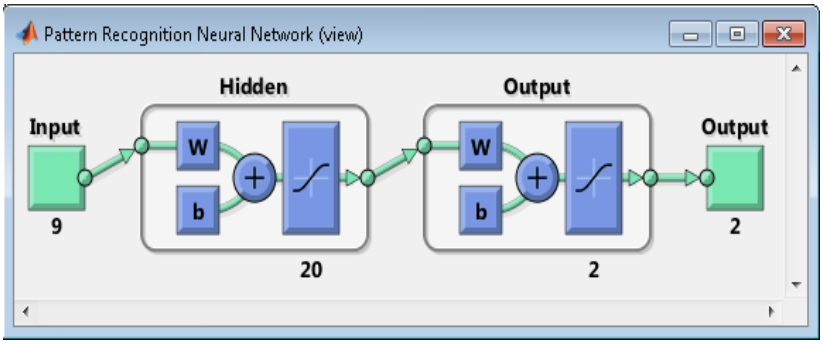


Figure 4.7: Simplified network diagram for automated event identification generated in MATLAB®. The network has *9* inputs, one hidden layer of size 20 with a sigmoid transfer function, and an ouput layer with *2* neurons and a sigmoid transfer function.

Layer 1 is a hidden layer. Layer 2 generates the network output so it is the ouput layer. The input to every intermediate layer is the output from its previous layer. The superscripts on all the network parameters indicate the layer to which they belong. Every input element to the ANN represents a certain feature of the detection signal. Usually, the input matrix consists of *p* (13488 in this case) sequences of number of inputs (9 in this case). Sequential inputs are converted into elements of a cell array before presenting them to the network. Layer 1 includes 9×p inputs, 20 neurons, a 20×9 weight matrix, 20×1 bias vectors, and a 20×p ouput vector. The output layer has 20×p inputs, 2 neurons, a 20×2 weight matrix, and a 2×p output marix. The training inputs and target matrices are

generated from the semi-automated collection of detection signals. The ouput of the ANN has same dimensions as that of the target matrix (2 in this case). A symmetric sigmoid transfer function, as shown in Figure 4.9, is used in both the hidden and output layers since it is easily differentiable, maps to the distribution of target values, and has a bound range between -1 and 1.

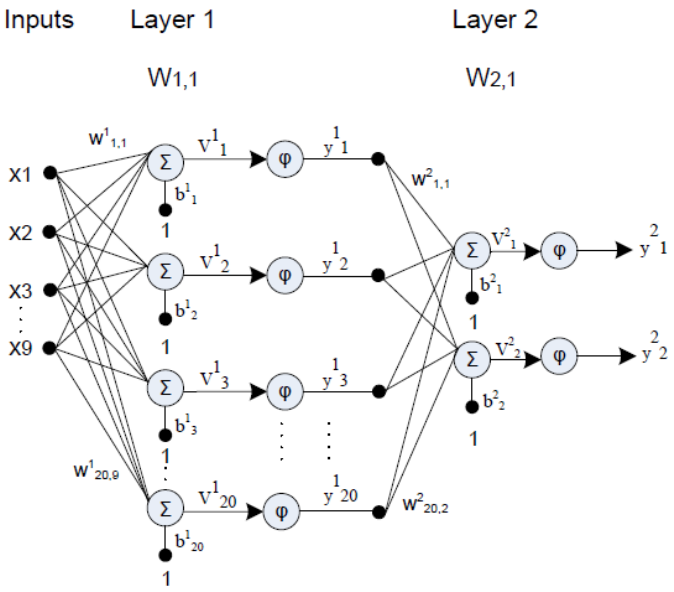


Figure 4.8: Two-layered feed-forward ANN for automated event identification.

The outputs of layers 1 and 2 are expressed in equations 4.13a, 4.13b and 4.13c.

$$y^1 = \varphi^1(W_{1,1}x + b^1) \tag{4.13a}$$

$$y^2 = \varphi^2(W_{2,1}y^1 + b^2) \tag{4.13b}$$

$$y^2 = \varphi^2(W_{2,1}(\varphi^1(W_{1,1}x + b^1)) + b^2 \tag{4.13c}$$

where $\varphi(\cdot) = \frac{1-e^{-v}}{1+e^{v}}$ is the expression for a symmetric sigmoid transfer function.

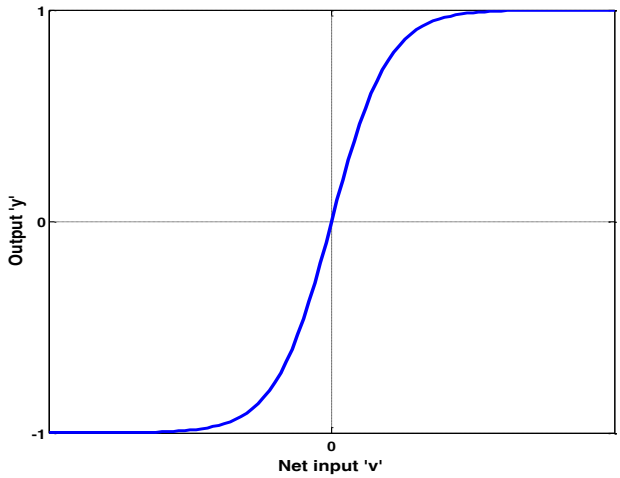


Figure 4.9: Symmetric Sigmoid function takes each element of the net input *v* in the interval [-inf inf] and squashes it into the interval [-1 1].

4.6 Feature selection for training the ANN

One of the most important criterion in designing an ANN is selecting the appropriate features for training the ANN from a known set of various features. The performance of a network depends on the types of features and number of features fed to it as inputs. The real deciding factor of a network's performance is how effective it works in correctly recognizing unseen data. Selecting too many features increases the network size and its complexity while, selecting fewer features may lead to incorrect classification of valid data. Hence, the features have to be chosen in such a way that the network is less complex and still correctly recognizes data.

The input patterns (or training inputs) to be presented to the ANN for training are generated from the semi-automated collection of detection signals. The training targets are labeled as accept or reject depending on the user's decision to accept or reject a detection signal. Initially, the number of neurons in the hidden layers is set to 10 and the

appropriate features can be chosen by monitoring the confusion matrix. Once the suitable features are chosen, the size of the hidden layer is set by varying the number of neurons to find a number at which the network has the best performance (least error) and a lower percentage of misclassified data as explained in Section 4.5. The confusion matrix is another measure of the performance of an ANN containing percentages of valid patterns being recognized as valid, valid patterns being recognized as invalid, invalid patterns being recognized as valid, and invalid patterns being recognized as invalid. The percentage of correctly classified data constitutes valid patterns being recognized as valid and invalid patterns being recognized as invalid. Valid patterns classified as valid, categorized as 'Accept', is the data accepted by both the user and the network. Invalid patterns classified as invalid falling under the category 'Reject' and is the data rejected by the user and the network. The percentage of misclassified data includes valid patterns being recognized as invalid, and invalid patterns being recognized as valid. Valid patterns classified as invalid is the patterns accepted by the user but is rejected by the network. Invalid patterns classified as valid is the data rejected by the user but is accepted by the network. The characteristics/features of correctly and misclassified data is explained in detail in Section 4.7.

The features of a detection signal from which important information can be extracted are: 1) Number of peaks; 2) Number of crossings at threshold levels 4, 6 and 8; 3) Time indices of peaks; 4) Time difference between peak times; 5) Widths of peaks; 6) Force index; and 7) Average Velocity.

The number of peaks are calculated by counting the peak times falling in between the in and out- times of every peak. To calculate the number of crossings [42] at different

levels, the dectection signal is divided into 20 equal voltage steps (threshold levels) and the number of times a detection signal crosses threshold levels 4, 6, and 8 is counted. To calculate the time indices of the peak times, the detection signal is divided into 20 equal time bins and the time bin in which each peak occurs is noted. The widths W1 and W2 of peaks 1 and peak 2 are calculated from Equations 2.7a and 2.7b respectively. The normalized peak time difference Δt is calculated from Equation 2.8. Force index is calculated from Equation 2.1. The average velocity of a cell registering a detection signal is calculated from the Equation 2.9. Combinations of features examined before the suitable input patterns are chosen are tabulated in Table 4.1. The check mark (✓) indicates the features included in a particular combination. Percentages of correctly classified training data, misclassified training data and the performance of the network for the various feature combinations are monitored and tabulated in Table 4.2.

Table 4.1: Combinations of various features tested to select the appropriate features.

Feature	**Combination Number**								
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
No. of peaks	✓	✓	✓	✓	✓	✓	✓	✓	✓
No. of Crossings at level 4	-	✓	✓	✓	✓	✓	✓	✓	✓
No. of Crossings at level 6	-	-	✓	✓	✓	✓	✓	✓	✓
No. of Crossings at level 8	-	-	-	✓	✓	✓	✓	✓	✓
Time bin indices of Peak 1 and Peak 2	-	-	-	-	✓	✓	✓	✓	✓
Normalized time diff. between peaks	-	-	-	-	-	✓	✓	✓	✓

Force Index	-	-	-	-	-	-	-	✓	✓
Average Velocity	-	-	-	-	-	-	-	-	✓

Table 4.2: Percentages of correctly classified, misclassified data and the MSE to choose the best features.

Combination Number	**% of correctly classified training data**	**% of misclassified training data**	**Performance (MSE)**
1	96.6%	3.4%	0.0322
2	96.6%	3.5%	0.0310
3	96.4%	3.5%	0.0293
4	96.5%	3.5%	0.0296
5	96.6%	3.3%	0.0290
6	96.6%	3.4%	0.0276
7	96.9%	3.3%	0.0271
8	96.7%	3.4%	0.0314
9	23%	77.0%	0.4162

The features of detection signals indicated by combination 7 in Table 4.1 are chosen as training input patterns to be presented to the network for training. On presenting the features of combination 7 to the network, it generates the smallest error and has lowest percentage of misclassifiction among all other combinations tested.

4.7 Data Classification

4.7.1 Correctly Classified Data

The main aim of designing an ANN is to successfully seperate useful and unusable detection signals into two categories, viz., Accept and Reject respectively. Figure 4.10, shows an acceptable detection signal marked with the best chosen features. The vertical

gridlines show the division of the signal into 20 threshold levels to count the number of crossings at levels 4, 6 and 8. The horizontal gridlines show the division of the signal along the time domain to identify the time indices of both peaks. The ANN is trained to accept no *DEP*, *pDEP* and *nDEP* signals shown in Figure 4.11. Input patterns of the acceptable detection signals are tabulated in Table 4.3.

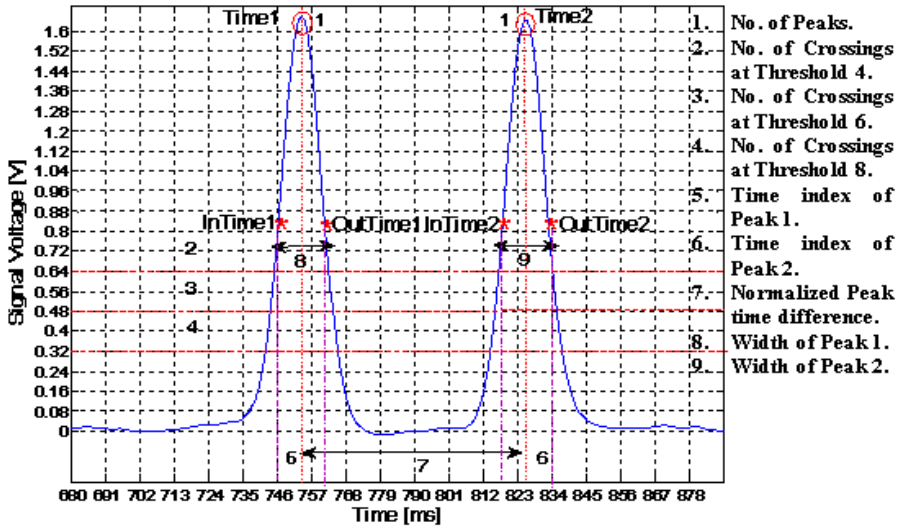


Figure 4.10: Accepted no *DEP* detection signals marked with all features.

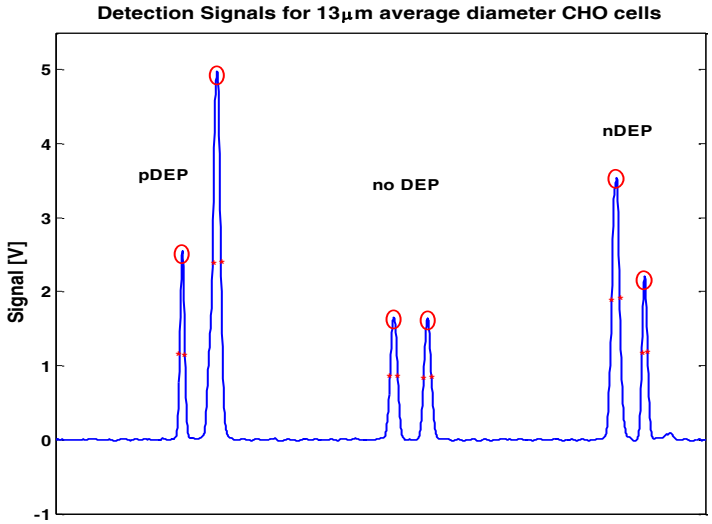


Figure 4.11: Detection signals accepted by the neural network.

Table 4.3: Patterns for the detection signals shown in Figure 4.11.

Feature no.	1	2	3	4	5	6	7	8	9
No ***DEP***	2	4	4	4	7	14	0.79	0.013	0.014
pDEP	2	4	4	4	7	14	0.84	0.021	0.034
nDEP	2	4	4	4	7	14	0.81	0.025	0.024

Signals rejected by the network have multiple peaks, more than four crossings, and multiple time indices. These signals are generated due to multiple cells passing simultaneously, lump of cells stuck to each other to form a lump, purges in the channel, and noise. Figure 4.12 shows a rejected signal with all features marked. Examples of rejected detection signals are shown in Figure 4.13 and their patterns are tabulated in Table 4.4.

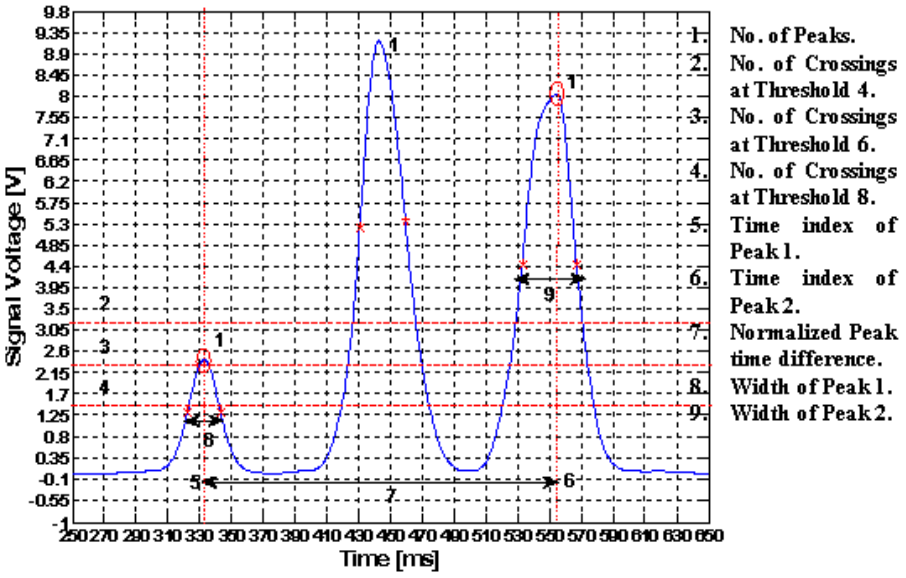


Figure 4.12: Rejected detection signal marked with all features.

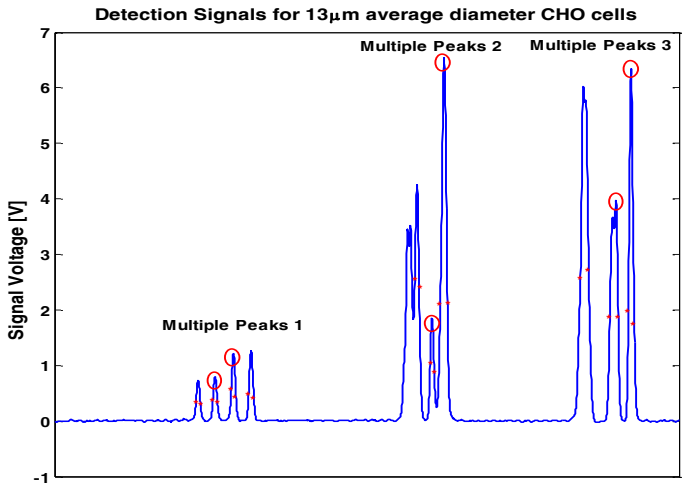


Figure 4.13: Detection signals rejected by the neural network.

Table 4.4: Patterns for the detection signals shown in Figure 4.14.

Feature no.	1	2	3	4	5	6	7	8	9
Multiple Peaks 1	4	8	8	8	5	8	0.86	0.031	0.026
Multiple Peaks 2	4	6	6	6	6	8	0.68	0.047	0.05
Multiple Peaks 3	3	6	6	6	6	12	0.83	0.026	0.027

As observed from Tables 4.3 and 4.4, accepted and rejected signals have distinctly different patterns based on which the ANN is trained to distinguish between them and assign separate categories to them.

4.7.2 Misclassified Data

Maintaing a low percentage of misclassified detection signals is very important for proper classification and to improve the efficieny of the ANN. One class of misclassified detection signals, as shown below, are originally rejected by the user during semi-automated collection but are accepted by the network; their patterns are tabulated in Table 4.5. Although the patterns of signals shown in Figure 4.16 are very similar to

patterns exhibited by acceptable signals tabulated in Table 4.3, they are rejected by the user for reasons explained further on.

Signal shown in Figure 4.14(a) is rejected by user because it does not reach the baseline between two peaks. Signals shown in Figures 4.14(b) and 4.14(c) are rejected because of the bumps in the signal due to two cells passing over the electrodes simultaneously. Peak 2 of the rejected signal shown in Figure 4.14(d) is saturated because of cell aggregation. Rejected signal shown in Figure 4.14(e) is rejected because it has a wider second peak resulting from the cell being stuck to the electrodes. The second peak of the signal shown in Figure 4.14(f) is not a result of actuation but due to a changing baseline.

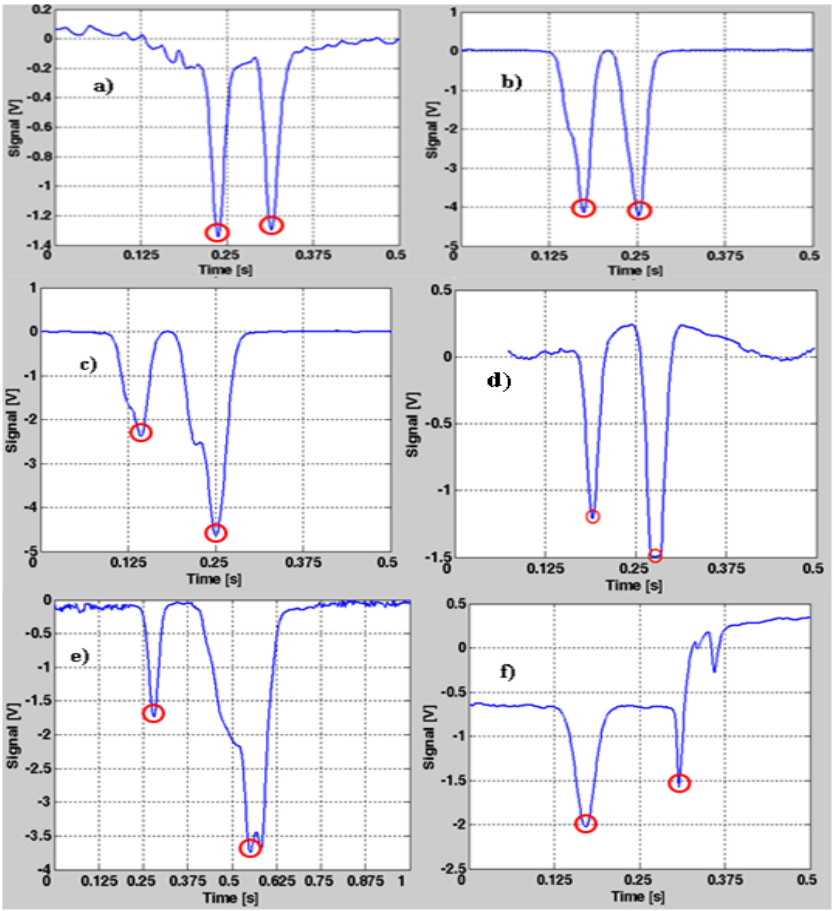


Figure 4.14: Misclassified detection signals: Rejected by the user during semi-automated collection, Accepted by the network.

Table 4.5: Patterns of misclassified signals shown in Figure 4.16.

Feature no.	1	2	3	4	5	6	7	8	9
Signal 4.16(a)	2	4	4	4	9	12	0.75	0.025	0.024
Signal 4.16(b)	2	4	4	4	9	12	0.70	0.031	0.031
Signal 4.16(c)	2	4	4	4	9	13	0.71	0.038	0.052
Signal 4.16(d)	2	4	4	4	8	12	0.83	0.02	0.03
Signal 4.16(e)	2	4	4	4	6	14	0.89	0.032	0.114
Signal 4.16(f)	4	6	8	10	8	12	-1.19	0.032	0.008

Another class of misclassified signals includes signals, as shown in Figure 4.15, are rejected by the network but accepted by the user. The patterns of these misclassified signals are tabulated in Table 4.6 and they clearly indicate that these signals are to be rejected. During semi-automated collection, these signals were to be rejected by the user but were accidently accepted as a result of the user clicking on the wrong button.

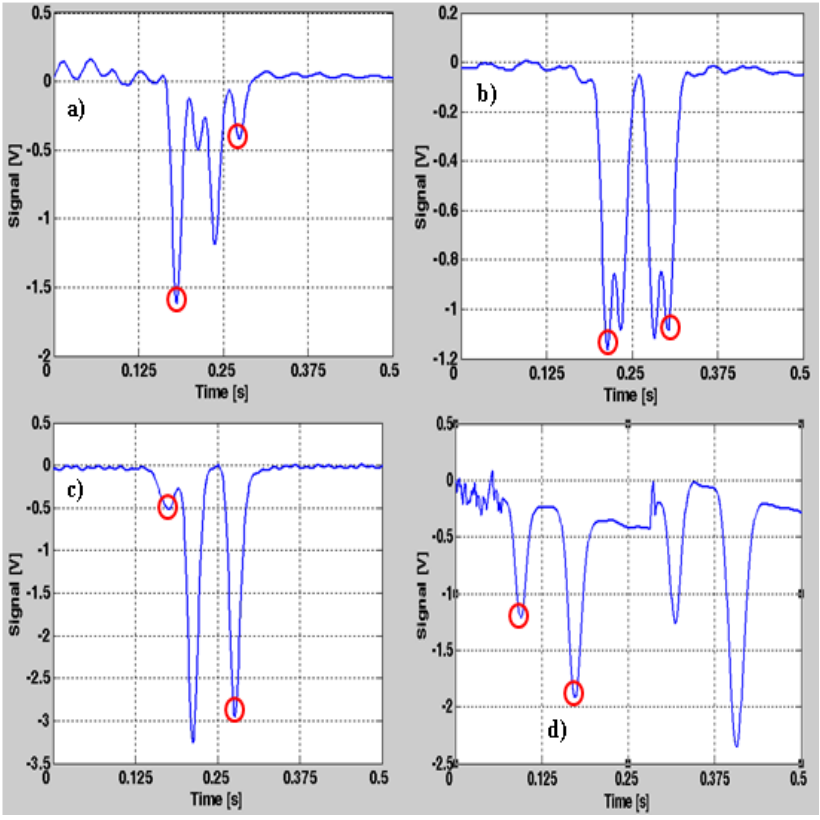


Figure 4.15: Misclassified detection signals: Accepted by the user during semi-automated collection, Rejected by the network.

Table 4.6: Patterns of misclassified signals shown in Figure 4.17.

Feature no.	1	2	3	4	5	6	7	8	9
Signal 4.17(a)	5	6	8	4	8	10	-1.24	0.017	0.018
Signal 4.17(b)	2	4	4	4	9	12	0.8	0.039	0.04
Signal 4.17(c)	3	6	4	4	7	9	1.6	0.029	0.019
Signal 4.17(d)	2	4	4	4	8	13	0.78	0.019	0.024

The ANN designed for classification of detection signals into 'Accept' and 'Reject' categories, as shown in Figure 4.8, is a two-layered feed-forward neural network with 9 inputs to the network, one hidden layer with 20 neurons, and 2 outputs. The training data presented to the network is a large set consisting of 13488 sample patterns. The network is trained by backpropagation algorithm based on scaled gradient descent optimization method. The confusion matrix, MSE, and regression analysis is an evaluation of the above designed network's performance and its efficieny in classifying detection signals into their appropriate categories.

The confusion matrix obtained after training the network is shown in Figure 4.16. In Figure 4.16, the green squares indicate the numbers and percentages of correctly classified data, the pink squares indicate the numbers and percentages of misclassified data, and the blue square shows the total percentages of correctly classified and misclassified data. In a confusion matrix '*C*', *C(1,1)* indicates the number and percentage of accepted signals being correctly classified as acceptable, *C(1,2)* indicates the number and percentage of accepted signals being misclassified as reject, *C(2,1)* indicates the number and percentage of rejected signals being misclassified as acceptable, and *C(2,2)* indicates the number and percentage of rejected signals being correctly classified as

reject. The confusion matrix shows that 96.9% of total number of samples (13064 out of 13488) are correctly classified while the remaining 3.1% are misclassifed (424 out of 13488). The percentage of misclassified signals is negligible in comparison to the percentage of correctly classified signals; misclassification is unavoidable due to the reasons mentioned in Section 4.7.2.

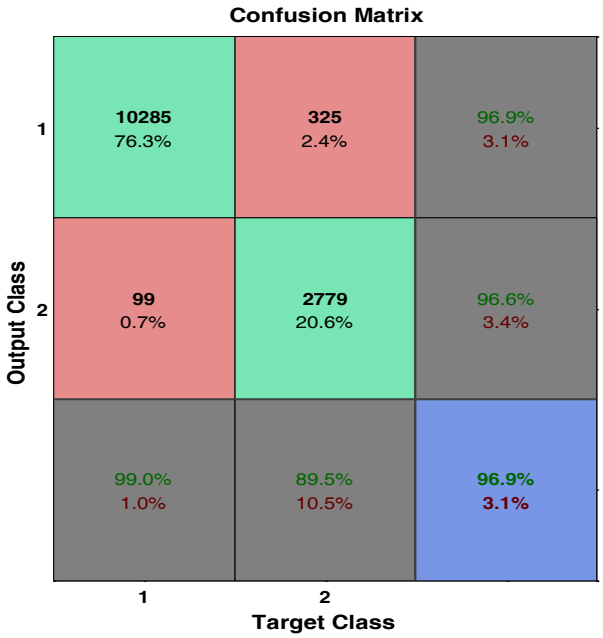


Figure 4.16: Confusion matrices for training, validation, and test data sets put together. 96.9% of the training data is correctly classified, 3.1% of the training data is misclassified.

Figure 4.17 shows that the performance function (MSE) follows the same curve for training, validation and test data sets. The validation data set monitors the network error and training stops when the error on the validation set is minimum [38]. Initially, the errors on both the training and the validation data sets decreases. The network begins to overfit the data when the validation error rises. The network's performance is the best at

112th iteration and the error on the validation set also reaches a minimum value at this iteration. The training continues for 6 more iterations before it is stopped. Figure 4.18 shows the relationship between targets and network outputs. The dashed line shows the ideal fit condition while the solid blue line shows the best linear fit between the targets and network outputs. The value R=0.96 indicates that there is a definite linear relationship between the targets and the outputs. Figures 4.16, 4.17 and 4.18 are a good indication that this network is well trained to classify unseen data with an acceptable percentage of misclassification.

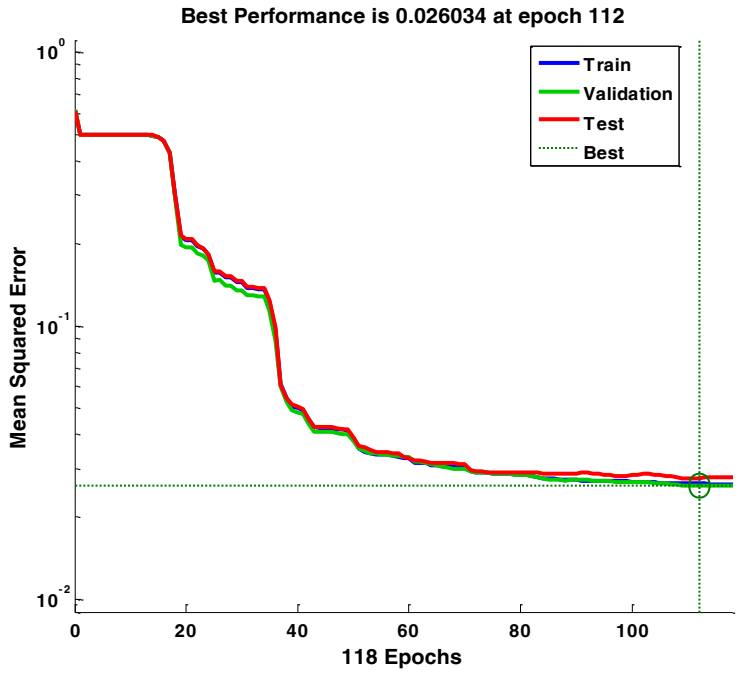


Figure 4.17: MSE vs. Iterations for training, validation, and test datasets. The network's performance is the best at 112th iteration. The error on the validation set (green line) reaches a minimum value at the 112th iteration.

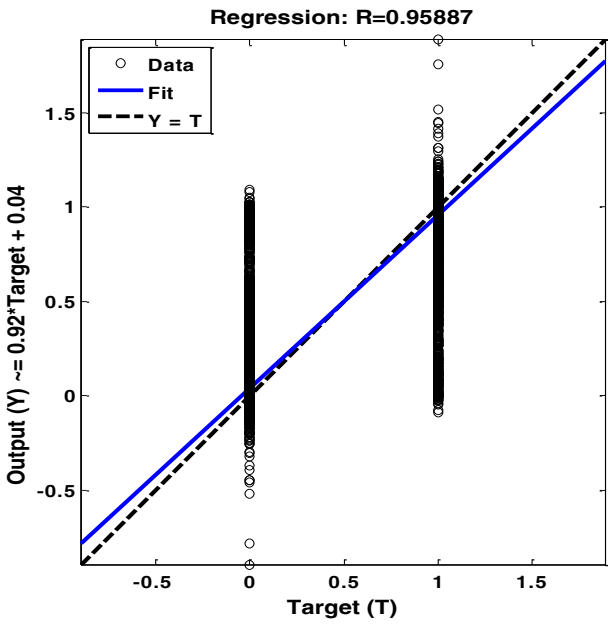


Figure 4.18: Regression plot for the designed network. R=0.96 indicates that there is a definite linear relationship between the targets and the outputs. The dashed line shows the ideal fit condition while the solid blue line shows the best linear fit between the targets and network outputs.

4.8 Post-Processing and Signal Analysis using the Automated Event Identifier

A MATLAB-based program was written to automatically classify detection signals and to analyze large volumes of real- time data in as little time as possible. The diagram shown in Figure 4.19, explains the flow of the program.

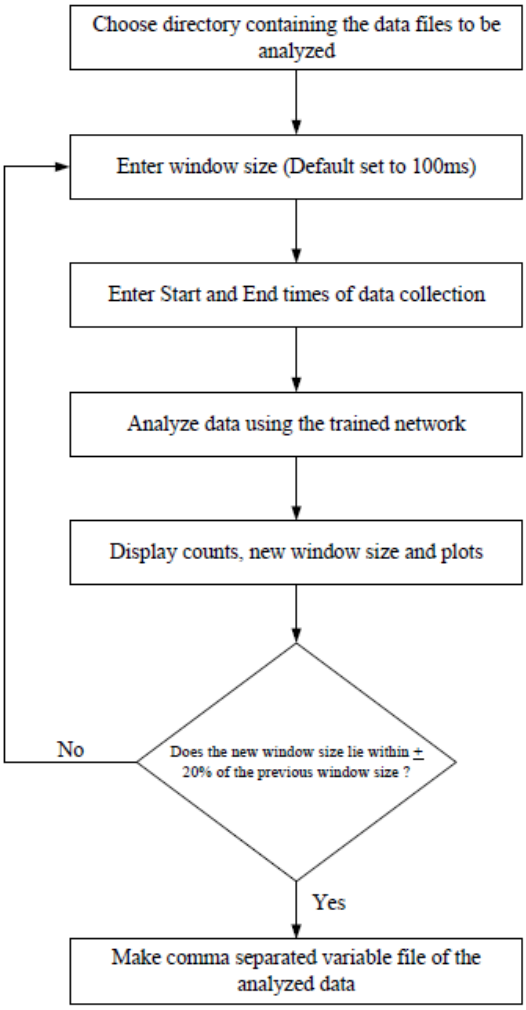


Figure 4.19: Flow diagram for the Automated Event identification program.

The program uses an interface, as shown in Figure 4.20, to display the results of data analysis.

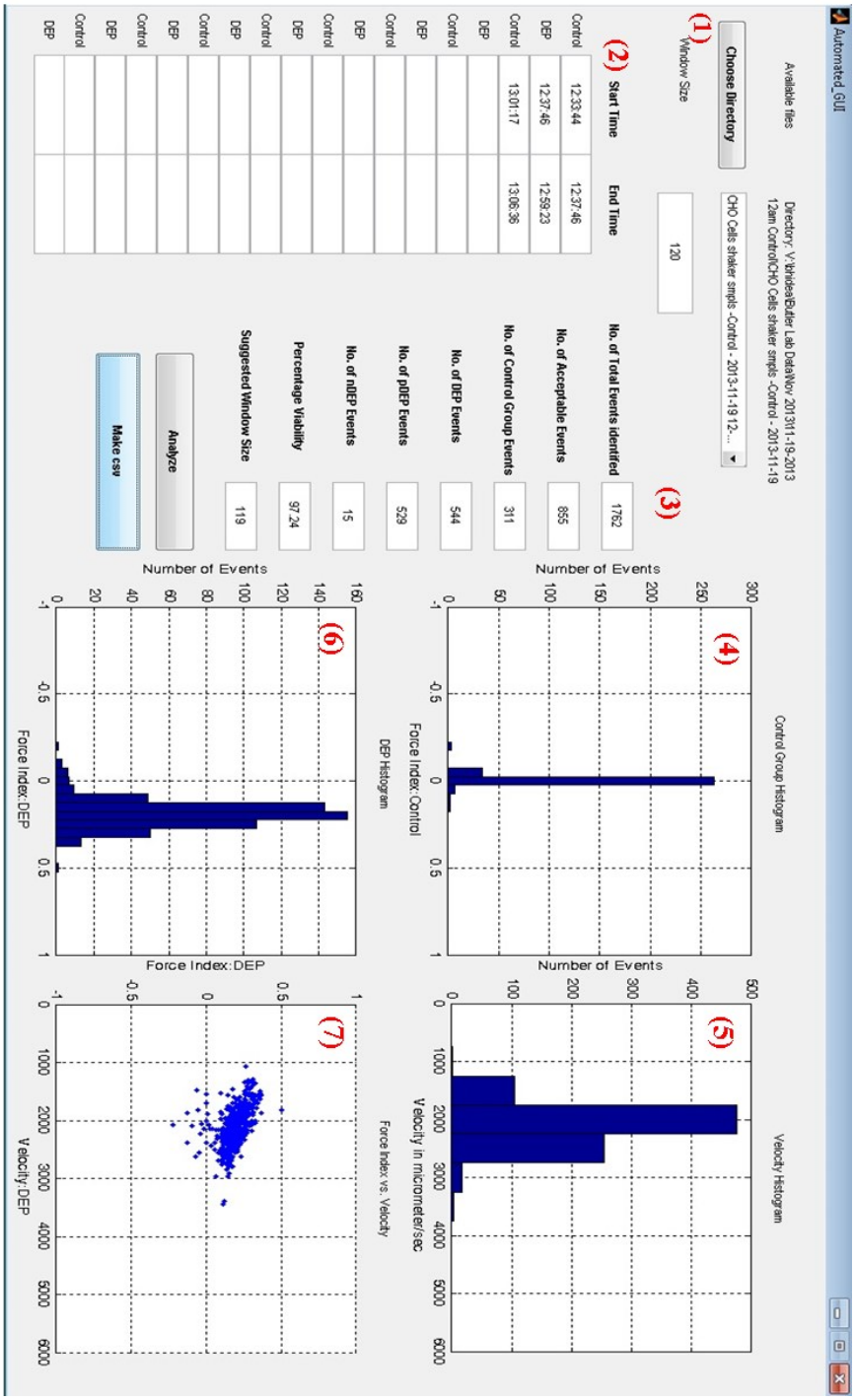


Figure 4.20: Post Processing using the Automated Event Identification program. (1) Box where the desired window size for data analysis is entered (2) Log book to enter the start and end times of collections (3) Counters that display the number of Control group

events, *DEP* events, percentage viability, and the suggested window size for the collection (4) Control group force index histogram (5) *DEP* force index histogram (6) Velocity histogram for *DEP* collection (7) Scatter plot of *DEP* force index vs. velocity plotted for every cell.

To start, the experimental data to be analyzed is located by clicking the "Choose Directory" button and cruising through the directory to the proper location of the data. The data files with the correct format are loaded by scanning the folder. The drop down menu "Available Files" is then populated with the data files from the experiment. The user can enter the desired window size in the edit box associated with the "Window size" text. The window size is the length of the rectangular window with which the data is filtered in the time domain to identify detection signals as events (Section 3.3.2). Default window size is set to 100 milliseconds (average time between peaks). In automated data collection, it is required to choose a correct window size such that it is wide enough to pick up detection signals from slow moving cells (v_{avg} >= 80μm/s) and small enough to pick signals from cells flowing at higher velocities (v_{avg} <= 5000μm/s). The importance of choosing a correct window size is discussed in Section 4.9.1. A log book is added to the interface for the user to enter the start and end times of control group and *DEP* data collections. The start times are entered under the header "Start Time" and the end times are entered under the header "End Time". A vertical column in the log book with tabs "Control" and "*DEP*" prompts the user to enter control group and *DEP* collection times in the spaces under the "Start Time" and "End Time" headers. On clicking the "Analyze" button, the program goes through various steps. First, the program loads one data file at a time in into the memory in a loop till the last data file. The loaded data file is then post-

processed by noise removal, filtering and detecting events as explained in Section 3.3.1 of Chapter 3.

The pattern of every detected event is generated and presented to the trained network to recognize acceptable detection signals and discard all other kinds of signals. The accepted detection signals are then separated into control group signals and *DEP* signals according to the times entered in the log book. Then, the counters added on the interface are updated and all plots are displayed. The counters display numbers of total signals, acceptable signals, control group signals, *DEP* signals and the percentage viability of the cell sample. Based on the force index values, *DEP* signals are separated into *pDEP* and *nDEP* signal counts. The "Suggested Window Size" displays a new window size depending on the average velocity of the cells. The new window size is calculated by subtracting the first standard deviation of the average velocity from the mean value of the average velocity. If the new window size falls within ±20% range of the previous window size, then data analysis ends else the window size is updated and the data is re-analyzed with new window size. The "Make csv" button enables the user to save the peak amplitudes, peak times, in-times, and out-times of acceptable signals, as shown in Figure 4.21, in two separate files: one file stores control group signals and the other file stores *DEP* signals. To the left of the interface, four plots are displayed: Control group distribution, *DEP* distribution, average velocity distribution, and a scatter plot of *DEP* force index values vs. average velocities.

1 Chip Name: IFSMB014 | Cell Type 96.8% via VCC 2.17e5 cells/ml | Buffer Type: Low Cond Media
2 RF Sensing Freq: 1.456GHz | RF Sensing Power: 13 dBm | Modulation Freq: 90.01kHz
3 LIA Time Const: 1ms | LIA Time Const Slope: 12 dB | LIA Sensitivity: 500 µV
4 Peak Revision
5 Start Time: 19-Nov-2013 12:27:00 | Sample Rate: 1000 Samples/s
6

7	In Time 1	Peak 1	Time 1	Out Time	T_Times1	In Time 2	Peak 2	Time 2	Out Time 2	T-Times2	Force Inde	Transition	# Crossing	# Crossing	# Crossing	#TimePea	#Time Bin	#Tine Bin 2
8	12:37:47.604	-1.83611	12:37:47.615	735235.5	0.021	12:37:47.716	-3.01479	12:37:47.734	12:37:47.744	0.028	0.242982	1.333333	4	4	4	2	7	14
9	12:37:49.781	-1.79293	12:37:49.792	735235.5	0.022	12:37:49.886	-2.79621	12:37:49.899	12:37:49.911	0.025	0.218621	1.136364	4	4	4	2	7	13
10	12:37:50.881	-1.70962	12:37:50.892	735235.5	0.022	12:37:50.986	-2.72056	12:37:50.998	12:37:51.011	0.025	0.228194	1.136364	4	4	4	2	7	13
11	12:37:52.657	-2.07903	12:37:52.668	735235.5	0.022	12:37:52.766	-3.52581	12:37:52.782	12:37:52.794	0.028	0.258131	1.272727	4	4	4	2	7	13
12	12:37:56.048	-1.73619	12:37:56.061	735235.5	0.024	12:37:56.167	-2.58083	12:37:56.181	12:37:56.194	0.027	0.195653	1.125	4	4	4	2	7	14
13	12:37:58.492	-2.63404	12:37:58.504	735235.5	0.023	12:37:58.614	-5.674	12:37:58.639	12:37:58.650	0.036	0.365905	1.565217	4	4	4	2	7	14
14	12:37:59.949	-2.6415	12:37:59.963	735235.5	0.027	12:38:00.097	-5.03621	12:38:00.126	12:38:00.138	0.041	0.311904	1.518519	4	4	4	2	6	14
15	12:38:00.468	-1.67416	12:38:00.480	735235.5	0.023	12:38:00.575	-2.3936	12:38:00.588	12:38:00.602	0.027	0.176865	1.173913	4	4	4	2	7	13
16	12:38:00.896	-1.70616	12:38:00.906	735235.5	0.021	12:38:01.000	-2.45463	12:38:01.013	12:38:01.025	0.025	0.179886	1.190476	4	4	4	2	7	13
17	12:38:01.313	-1.84118	12:38:01.327	735235.5	0.026	12:38:01.444	-3.17974	12:38:01.466	12:38:01.478	0.034	0.266598	1.307692	4	4	4	2	7	14
18	12:38:04.314	-1.68262	12:38:04.325	735235.5	0.021	12:38:04.420	-2.70007	12:38:04.434	12:38:04.446	0.026	0.232152	1.238095	4	4	4	2	7	13
19	12:38:05.535	-6.46261	12:38:05.549	735235.5	0.028	12:38:05.711	-10.9931	12:38:05.746	12:38:05.761	0.05	0.25954	1.785714	4	4	4	2	6	14
20	12:38:11.175	-1.99438	12:38:11.187	735235.5	0.025	12:38:11.300	-3.09292	12:38:11.317	12:38:11.330	0.03	0.215938	1.2	4	4	4	2	7	14
21	12:38:12.420	-3.42216	12:38:12.433	735235.5	0.024	12:38:12.545	-6.17767	12:38:12.568	12:38:12.579	0.034	0.287038	1.416667	4	4	4	2	7	14
22	12:38:14.126	-3.25641	12:38:14.139	735235.5	0.025	12:38:14.252	-6.89928	12:38:14.268	12:38:14.286	0.034	0.358702	1.36	4	4	4	2	7	13
23	12:38:15.050	-1.92972	12:38:15.061	735235.5	0.022	12:38:15.158	-2.89686	12:38:15.171	12:38:15.184	0.026	0.200378	1.181818	4	4	4	2	7	13
24	12:38:17.271	-2.25326	12:38:17.282	735235.5	0.021	12:38:17.382	-4.20447	12:38:17.401	12:38:17.412	0.03	0.302151	1.428571	4	4	4	2	7	13
25	12:38:19.604	-2.35477	12:38:19.618	735235.5	0.027	12:38:19.721	-2.0938	12:38:19.733	12:38:19.747	0.026	-0.05867	0.962963	4	4	4	2	7	14
26	12:38:23.987	-2.80945	12:38:23.998	735235.5	0.022	12:38:24.101	-4.91855	12:38:24.121	12:38:24.133	0.032	0.272917	1.454545	4	4	4	2	7	13
27	12:38:25.135	-2.2928	12:38:25.147	735235.5	0.023	12:38:25.251	-3.45522	12:38:25.267	12:38:25.279	0.028	0.202229	1.217391	4	4	4	2	7	14
28	12:38:25.741	-1.76959	12:38:25.754	735235.5	0.025	12:38:25.879	-3.34381	12:38:25.904	12:38:25.916	0.037	0.30786	1.48	4	4	4	2	7	14
29	12:38:28.969	-1.64613	12:38:28.982	735235.5	0.023	12:38:29.083	-2.36515	12:38:29.098	12:38:29.110	0.027	0.17925	1.173913	4	4	4	2	7	14
30	12:38:29.656	-2.2599	12:38:29.669	735235.5	0.024	12:38:29.783	-4.01498	12:38:29.804	12:38:29.816	0.033	0.279699	1.375	4	4	4	2	7	14
31	12:38:33.837	-2.09532	12:38:33.850	735235.5	0.025	12:38:33.956	-3.21633	12:38:33.970	12:38:33.984	0.028	0.211047	1.12	4	4	4	2	7	14
32	12:38:39.267	-1.80504	12:38:39.278	735235.5	0.023	12:38:39.385	-2.91485	12:38:39.402	12:38:39.414	0.029	0.235136	1.26087	4	4	4	2	7	14
33	12:38:46.661	-2.53702	12:38:46.673	735235.5	0.023	12:38:46.772	-4.4104	12:38:46.789	12:38:46.800	0.028	0.269651	1.217391	4	4	4	2	7	13
34	12:38:47.297	-2.62764	12:38:47.311	735235.5	0.027	12:38:47.443	-4.66265	12:38:47.471	12:38:47.483	0.04	0.27914	1.481481	4	4	4	2	7	14
35	12:38:50.178	-2.04427	12:38:50.191	735235.5	0.027	12:38:50.322	-3.4258	12:38:50.348	12:38:50.360	0.038	0.25256	1.407407	4	4	4	2	7	14
36	12:38:54.399	-1.83541	12:38:54.410	735235.5	0.022	12:38:54.514	-2.873	12:38:54.530	12:38:54.543	0.029	0.22037	1.318182	4	4	4	2	7	14
37	12:38:55.265	-2.20684	12:38:55.278	735235.5	0.024	12:38:55.385	-2.93152	12:38:55.398	12:38:55.412	0.027	0.141033	1.125	4	4	4	2	7	14
38	12:38:55.884	-1.89432	12:38:55.897	735235.5	0.026	12:38:56.021	-3.28835	12:38:56.046	12:38:56.058	0.037	0.26898	1.423077	4	4	4	2	7	14

Figure 4.21: CSV file containing the processed data collected using the Automated Event Identification program.

4.9 Results and Conclusions

Manual data collection gives a user the freedom to pick useful signals that makes it a reliable benchmark with which other signals analysis methods can be compared. The semi-automated data collection method is compared with the manual data collection method in Chapter 3. The results show that the semi-automated event identifier is quite efficient in identifying events with a small percentage of loss in comparison to the manual data collection method. The automated event identifier has evolved from the semi-automated event identification system. Therefore, it is useful to compare the automated data collection method with both the manual and semi-automated data collection methods to ensure that reduction in signal analysis time does not in any way alter the statistics of the collected data. The affect of change in window sizes on the statistics for data collected by the automated method is also reported.

4.9.1 Comparing Results for Different Window Sizes

In this section, results for data collected with different window sizes by semi-automated and automated methods are presented. For the semi-automated method, data is analyzed using a dynamic window size such that only one detection signal is identified at a time. In the semi-automated collection, the window size can be changed by the user during data collection. For the automated method, the window size is initially set to 100ms and then varied depending on the suggested window size. Figures 4.22 and 4.23 show the comparison of the number of control and *DEP* detection signals identified for different window sizes by semi-automated and automated data collection methods. The optimum window size for this particular data set is 75ms. A very small window size picks

up only half a detection signal while a very large window size picks up more than one detection signal; the neural network is trained to reject such signals. Figures 4.22 and 4.23 show that the number of signals identified begins to drop down when the window size is increased beyond 125ms. The number of events drop down for a 175ms window size and rise again for a 200ms window size. A 175ms window size is small to identify events that are generated by slow moving cells (control group) or cells experiencing a very strong *DEP* force. A line is plotted for semi-automated collection only to show that at 75ms window size, the number of events identified by automated data collection is equal to the number of events identified by semi-automated data collection.

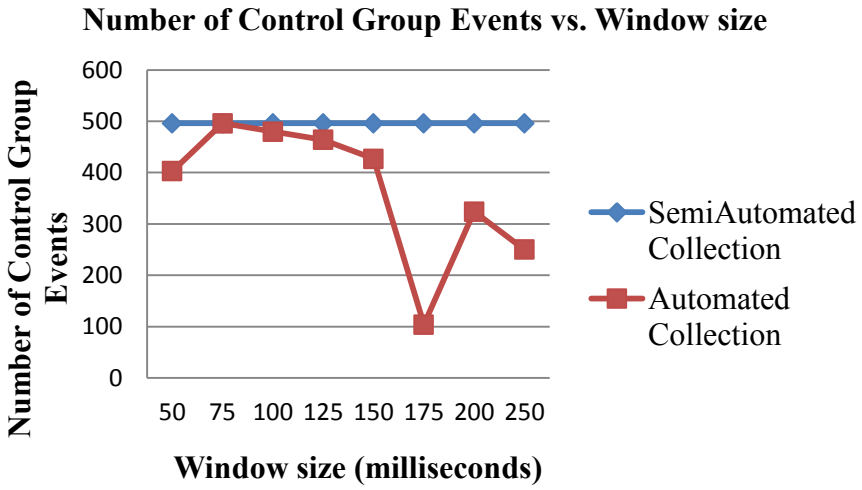


Figure 4.22: Comparison of number of control group signals identified with change in window size for semi-automated and automated collections.

The loss percentage is defined as the number of signals (control group or *DEP*) lost in automated collection in comparison to the semi-automated collection for window sizes in the range 50ms-250ms. The loss percentage is calculated by Equation 4.14.

$$Loss\ percentage = \frac{(Number\ of\ signals\ identified\ by\ semiautomated\ collection - Number\ of\ signals\ identified\ by\ automated\ collection) * 100}{Number\ of\ signals\ identified\ by\ semiautomated\ collection} \quad (4.14)$$

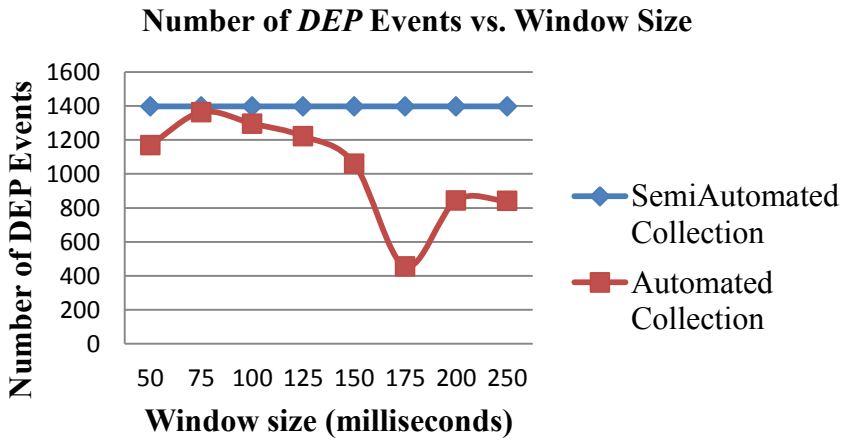


Figure 4.23: Comparison of number of *DEP* signals identified with change in window size for semi-automated and automated collections.

The percentage of loss in the number of events identified by automated data collection in comparison to the number of events identified by semi-automated data collection is shown in Figure 4.24. The solid blue line shows the curve for percentage loss in identifying control group events when the window size is varied from 50ms to 250ms. The solid red line shows the curve for percentage loss in identifying *DEP* events for varying the window size from 50ms to 250ms. The mean force index values for control group distributions lie around zero for all the window sizes in the range 50ms-250ms; control group force index values do not contribute in calculating the viability percentages so comparing statistics for control group force index values is not required. However, the control group distributions for different window sizes are shown further in this chapter.

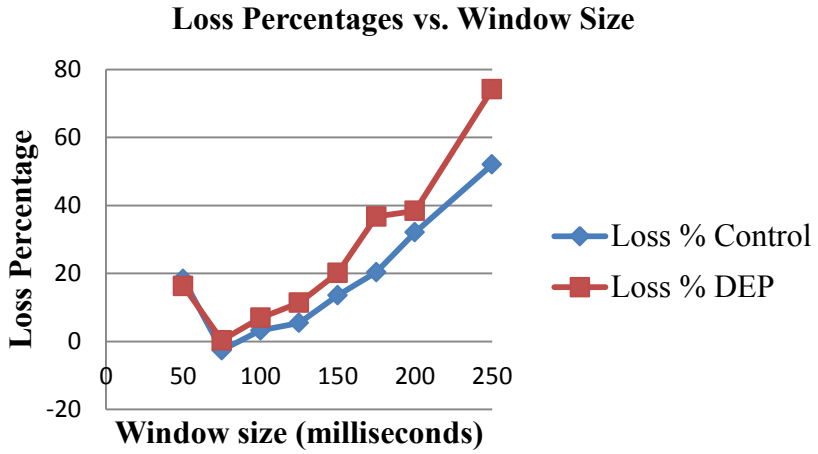


Figure 4.24: Loss percentages in control group and *DEP* Events with change in window sizes for automated data collection.

The mean force index values for *DEP* distributions obtained for different window sizes are compared for semi-automated and automated data collections as shown in Figure 4.25. In Figure 4.26, the deviations in the mean force index values of the *DEP* distributions obtained form varying window sizes from 50ms to 250ms are compared for both semi-automated and automated data collections. At 75ms, the mean force index and the deviation of the force index from its mean value for both the data collections overlap.

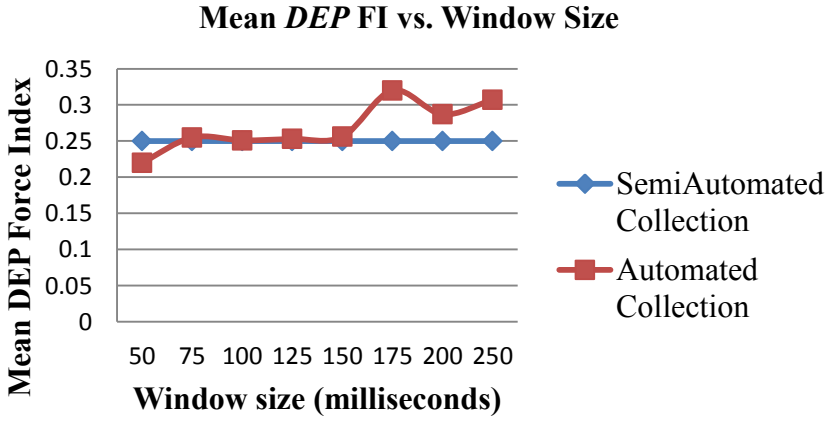


Figure 4.25: Comparison of mean force index values for *DEP* distributions obtained by varying window sizes for semi-automated and automated collections.

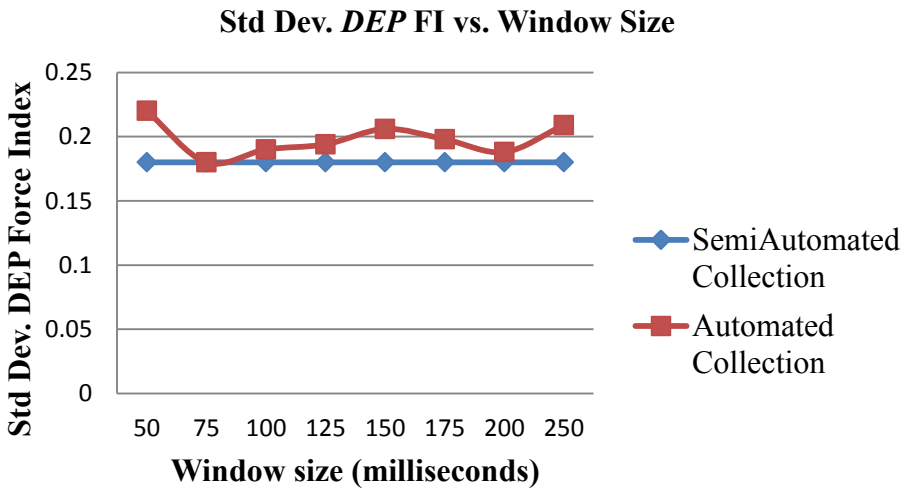


Figure 4.26: Comparison of standard deviations for *DEP* distributions obtained by varying window sizes for semi-automated and automated collections.

The distributions of control group and *DEP* events obtained by varying the window sizes from 50ms to 250ms for automated data collection are compared to the distributions of events identified by the semi-automated data collection. The comparison of force index distributions obtained for both the data collections at different window sizes are shown in Figures 4.27, 4.28, 4.29, 4.30, 4.31, 4.32, 4.32, 4.33, and 4.34. The distributions in blue are generated from the semi-automated collection of data with a dynamic window size while the distributions in red are generated from the automated collection of data with a fixed window size.

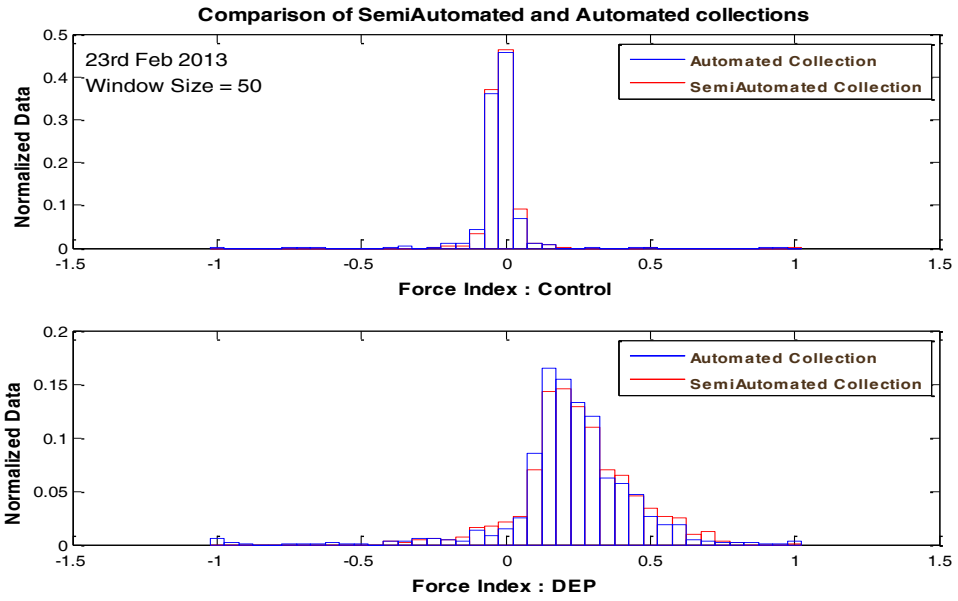


Figure 4.27: Comparison of force index distributions for semi-automated and automated data collections for 50ms window size.

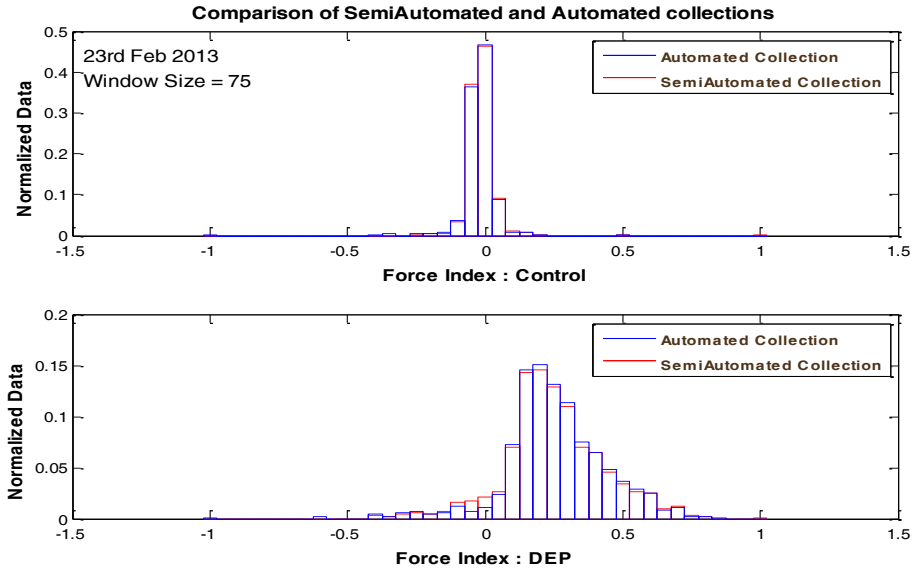


Figure 4.28: Comparison of force index distributions for semi-automated and automated data collections for 75ms window size.

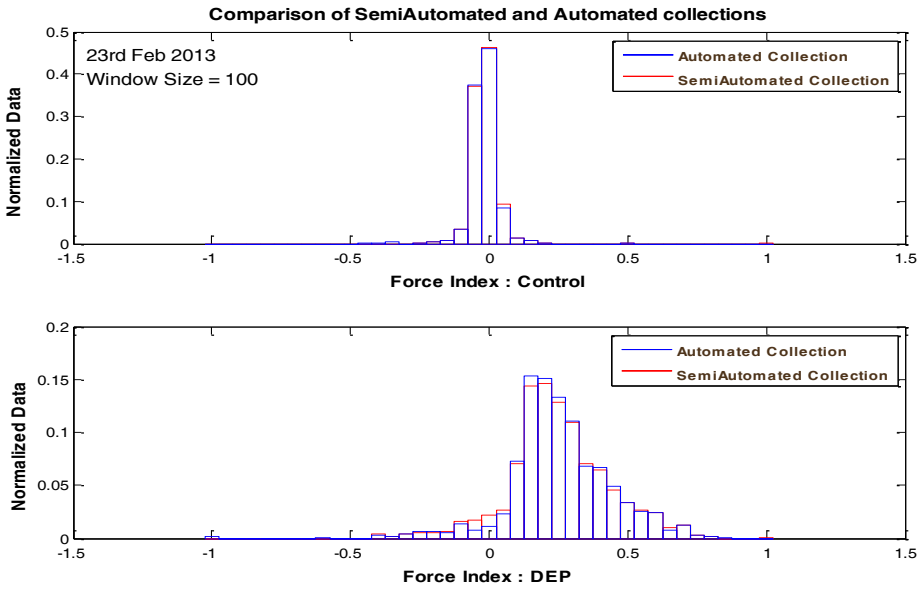


Figure 4.29: Comparison of force index distributions for semi-automated and automated data collections for 100ms window size.

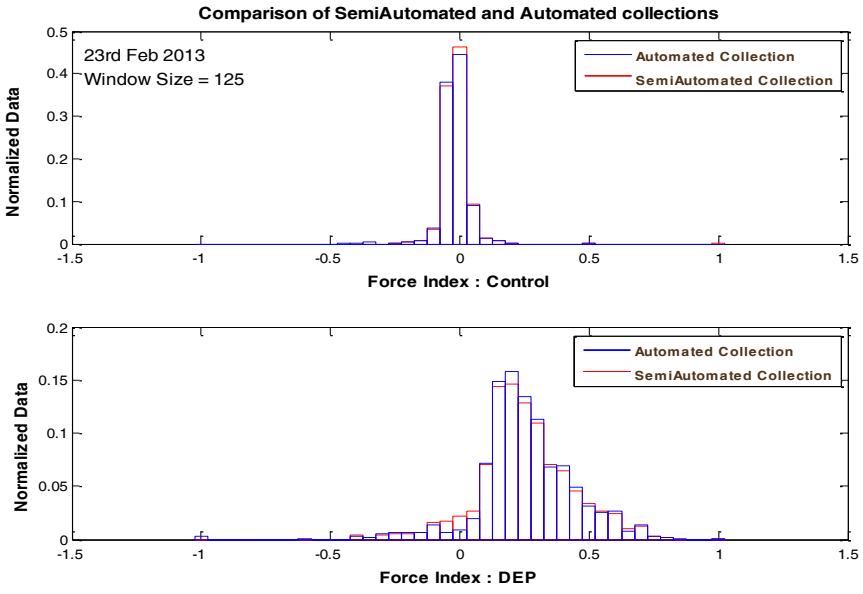


Figure 4.30: Comparison of force index distributions for semi-automated and automated data collections for 125ms window size.

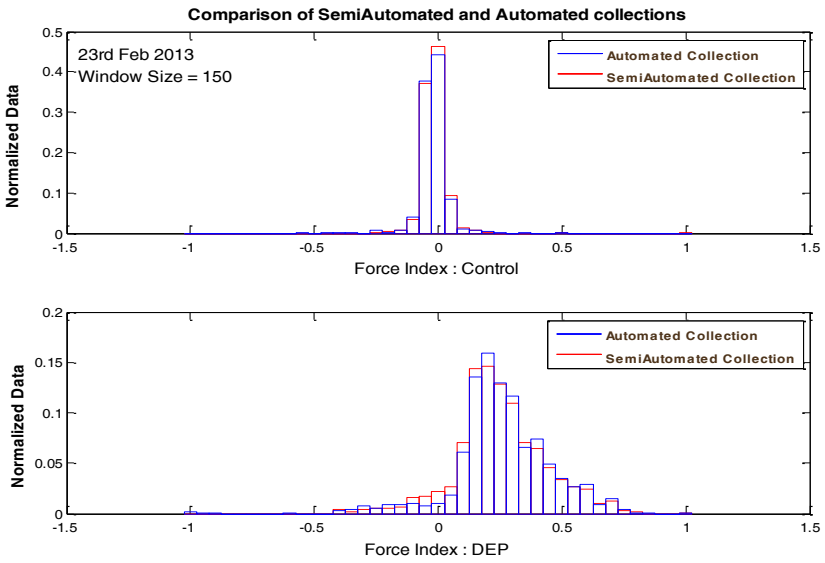


Figure 4.31: Comparison of force index distributions for semi-automated and automated data collections for 150ms window size.

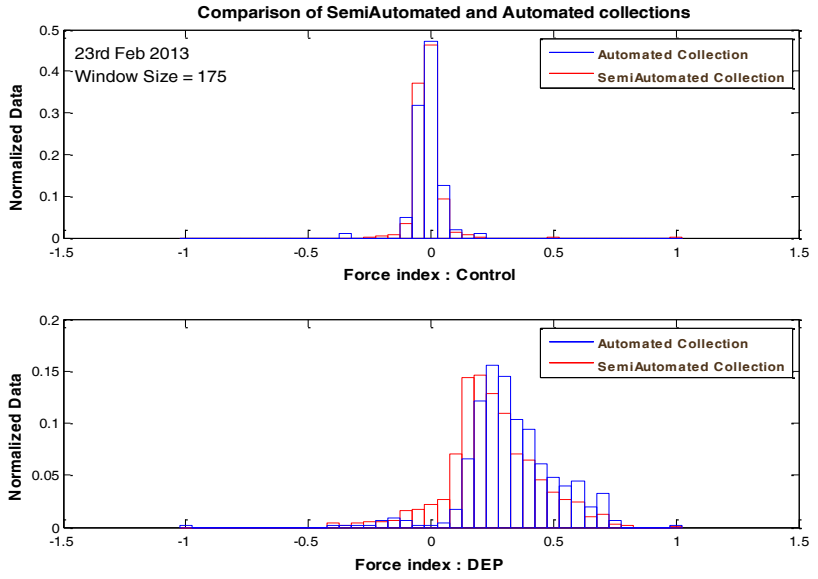


Figure 4.32: Comparison of force index distributions for semi-automated and automated data collections for 175ms window size.

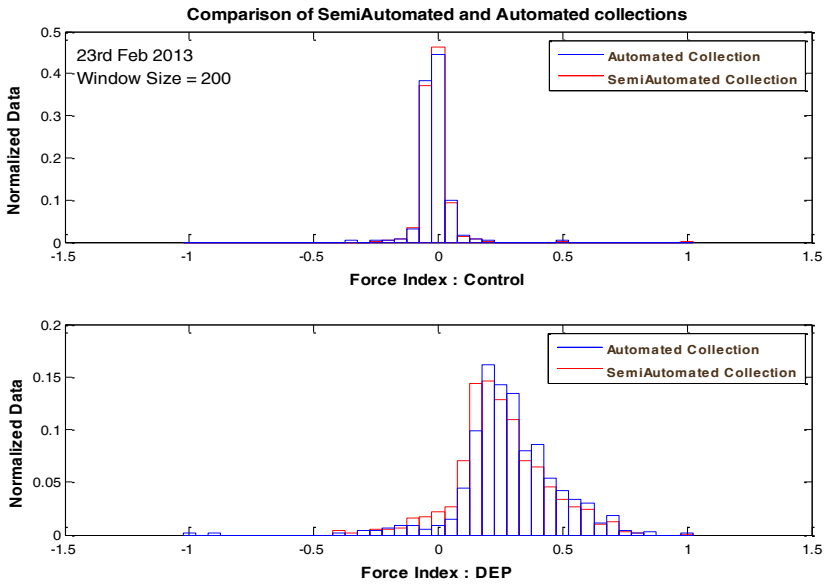


Figure 4.33: Comparison of force index distributions for semi-automated and automated data collections for 200ms window size.

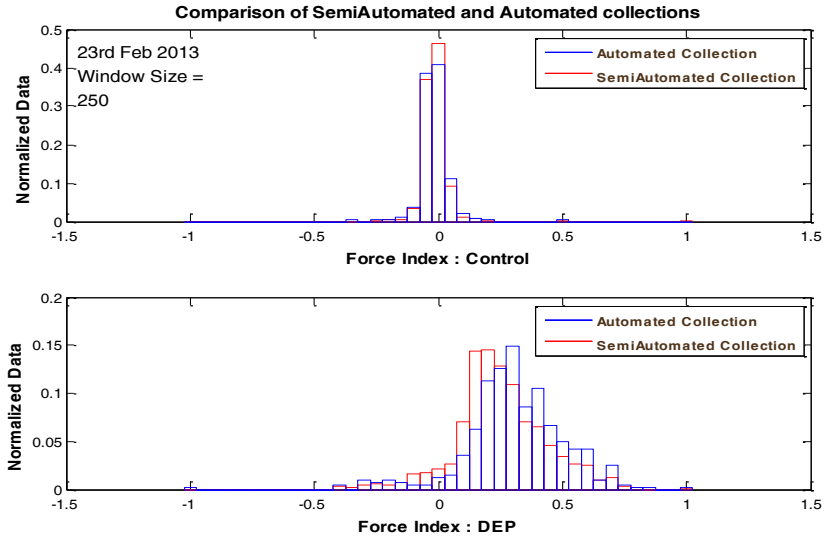


Figure 4.34: Comparison of force index distributions for semi-automated and automated data collections for 250ms window size.

Among all distributions shown above, the best overlap between both the collections is found to be at 75ms. As the window size increases, it is evident that the number of *DEP* events identified by the automated data collection method start to decline. The mean values of the distributions shift away from the mean values obtained from the semi-automated collection.

Table 4.7: *T*-test results performed on data analyzed with different window sizes by the automated method and the same data analyzed by a dynamic window size using the semi-automated method. The *T*-tests indicate that the best window sizes for automated signal classification are between 75ms-125ms.

Window Size	**Hypothesis Test Result '*H*'**		***p*-value**	
	Control	***DEP***	**Control**	***DEP***
50	1	1	0.0138	0.0016
75	0	0	0.085	0.318
100	0	0	0.2	0.76
125	0	0	0.114	0.514
150	0	1	0.094	0.042
175	0	1	0.318	0.0135
200	0	1	0.548	1.1e-4
250	0	1	0.629	7.02e-9

T-tests are performed on data analyzed with different window sizes collected by the automated method and the same data analyzed by a dynamic window size collected by the semi-automated method. The results of the *T*-tests, as tabulated in Table 4.7, show that the null hypothesis for both control group and *DEP* data stands true (*H=0*) only for window sizes 75ms, 100ms and 125ms. The *T*-test results also indicate that the best window sizes for automated signal classification are between 75ms-125ms.

4.9.2 Comparing Semi-Automated and Automated Methods

For an experiment conducted to measure the dielectric properties of CHO cells in the frequency range 50 KHz-20MHz, the electronically acquired detection signals are analyzed using both semi-automated and automated techniques. The results obtained from the automated collection of data are compared with the results obtained from the semi-automated data collection. The normalized distributions of control group data for semi-automated and automated collections are shown in Figure 4.35. The distribution in solid red lines is obtained from the semi-automated method while the distribution in blue lines is from the automated data collection method. Uncertainties in the number of events are calculated from Equation 3.1. As shown in Figure 4.36, for both types of data collections, the uncertainties in the number of events for every force index bin lie within the same range. Uncertainties are plotted only for force index bins having 10 or more events. For automated data collection, the window size is calculated by first obtaining an estimate of the average flow velocities. For this experiment, the data is analyzed for a window size of 150ms.

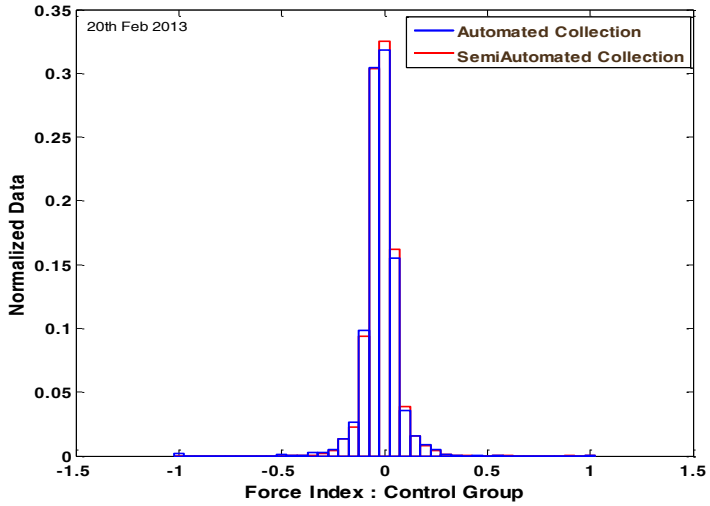


Figure 4.35: Comparison of control group force index distributions for semi-automated and automated collections.

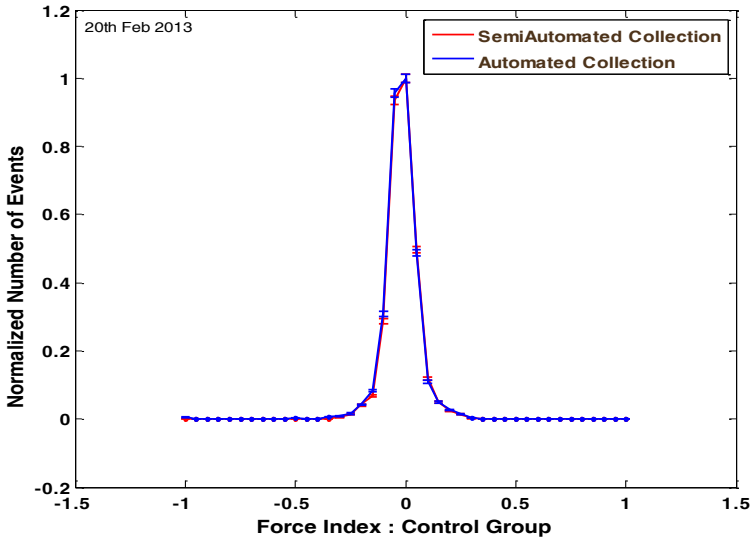


Figure 4.36: Uncertainty comparison in number of control group events for semi-automated and automated collections.

The normalized distributions of *DEP* data for semi-automated and automated collections at frequencies 50KHz, 200KHz, 400KHz, 600KHz, 800KHz, 1MHz, 2MHz, 4MHz, 6MHz, and 20MHz are shown in Figures 4.37, 4.38, 4.39, 4.40, and 4.41

respectively. The mean force index values for both collections follow the same curve, as shown in Figure 4.42, and the mean force index at crossover frequency (400 KHz) is 0.02 for both types of data collections. The comparisons for uncertainties in the number of *DEP* events at different frequencies are shown in Figures 4.43, 4.44, 4.45, 4.46, and 4.47.

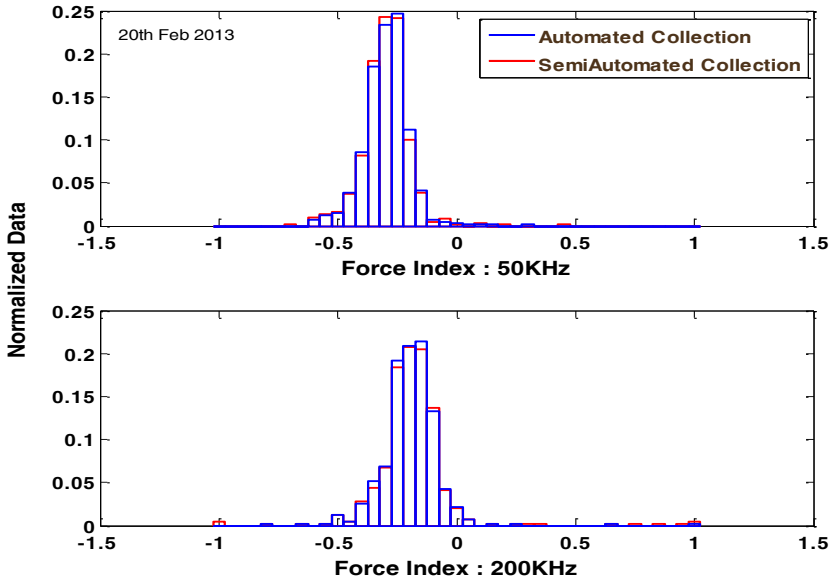


Figure 4.37: Comparison of *DEP* force index distributions at 50 KHz and 200 KHz for semi-automated and automated collections.

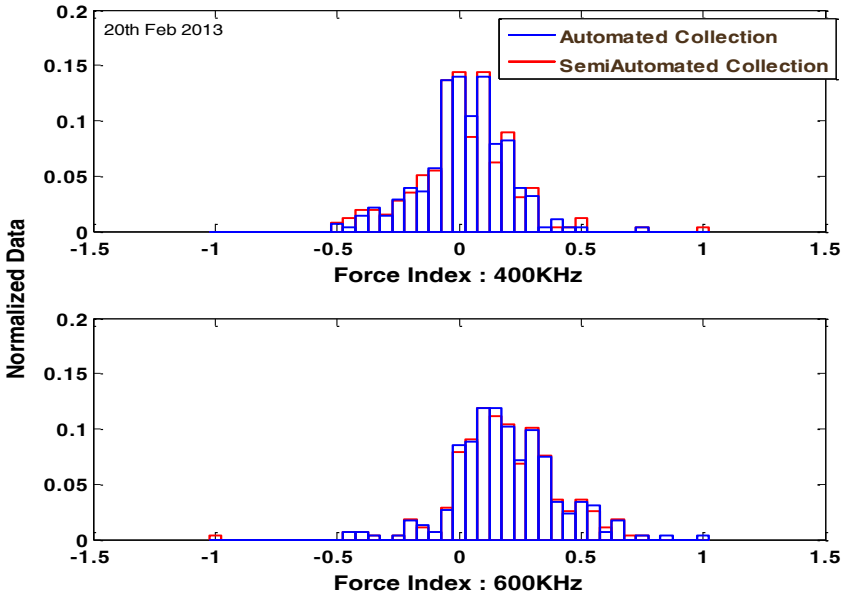


Figure 4.38: Comparison of *DEP* force index distributions at 400 KHz and 600 KHz for semi-automated and automated collections.

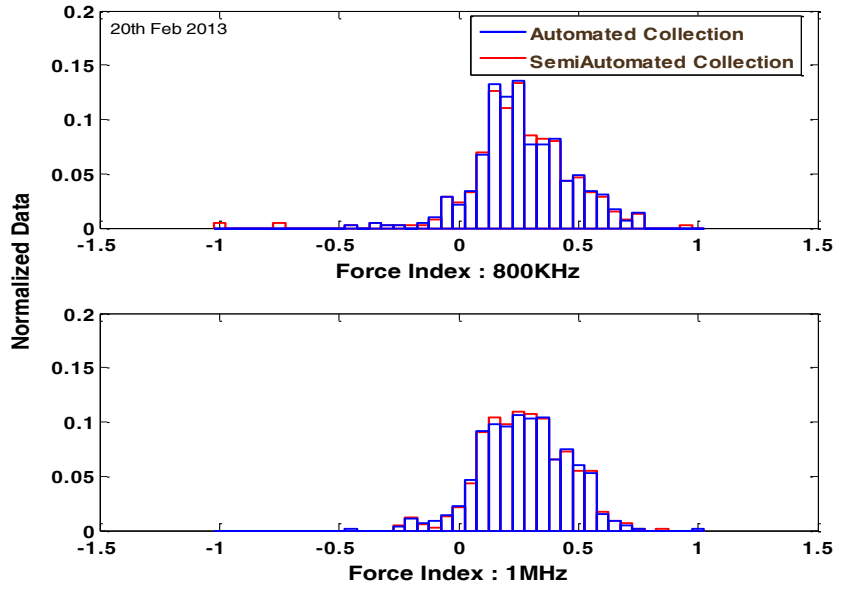


Figure 4.39: Comparison of *DEP* force index distributions at 800 KHz and 1 MHz for semi-automated and automated collections.

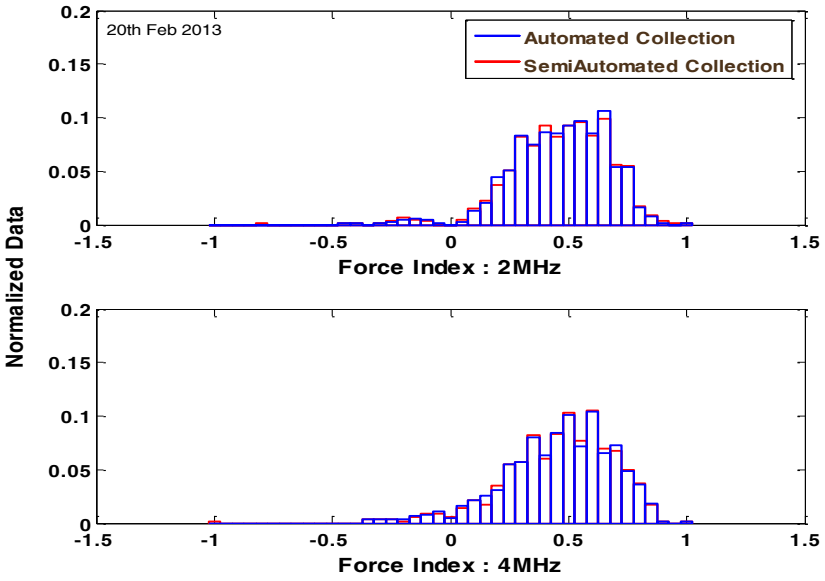


Figure 4.40: Comparison of *DEP* force index distributions at 2 MHz and 4 MHz for semi-automated and automated collections.

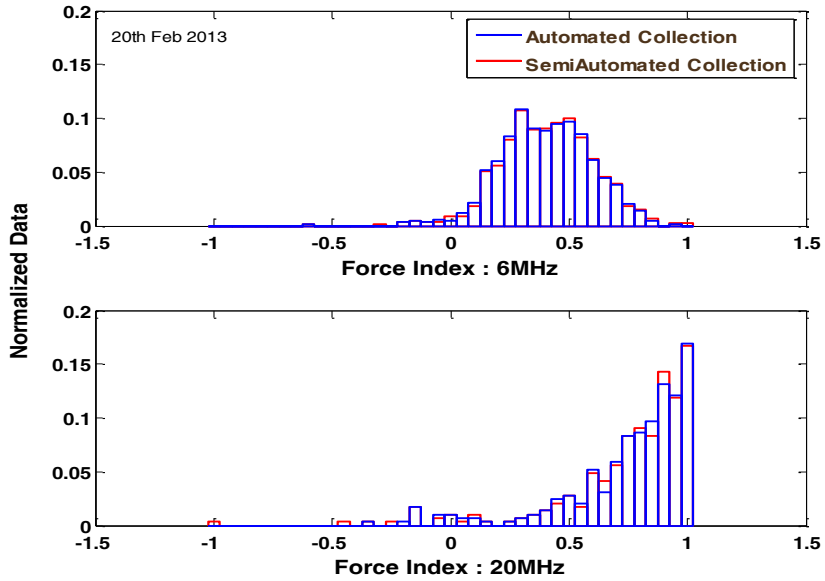


Figure 4.41: Comparison of *DEP* force index distributions at 6MHz and 20MHz for semi-automated and automated collections.

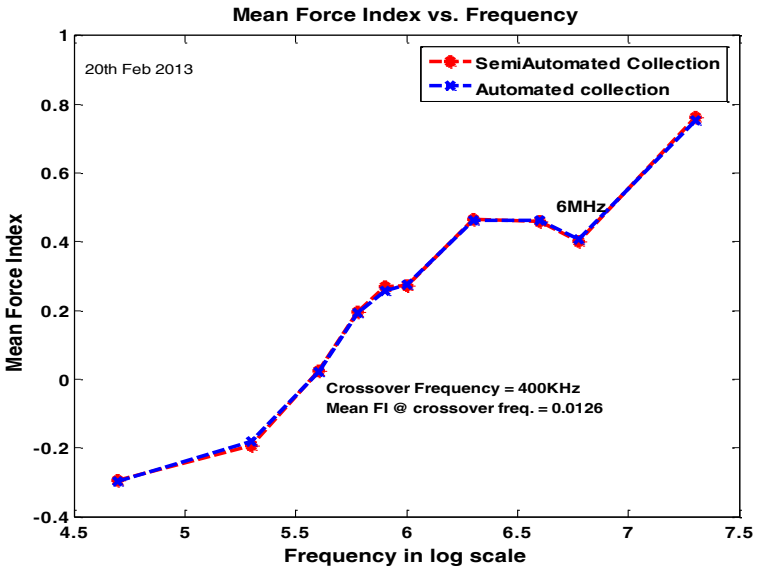


Figure 4.42: Comparison of mean force index values over the log frequency range 50 KHz-20 MHz for semi-automated and automated data collections. The change of mean force index from negative to positive occurs at 400 KHz.

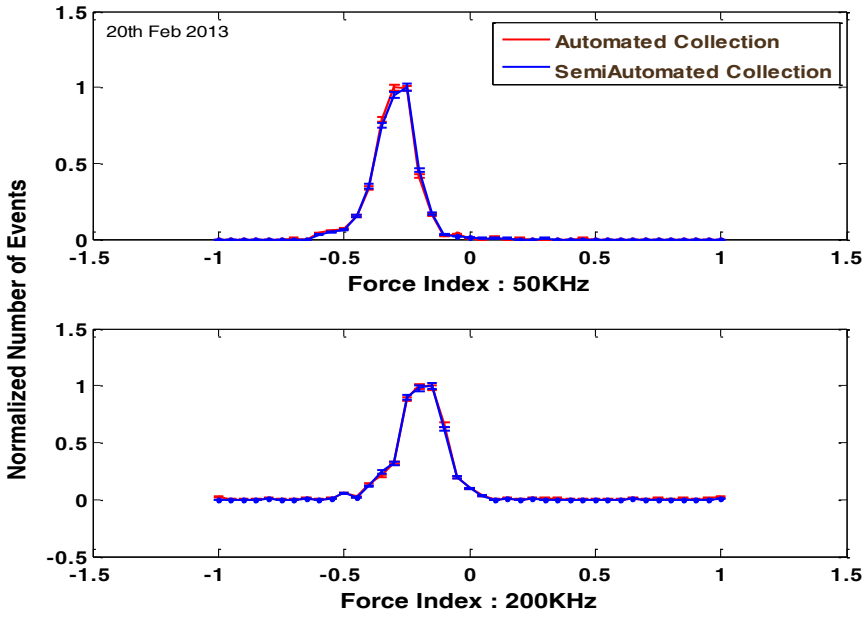


Figure 4.43: Uncertainty comparison in number of *DEP* events at 50 KHz and 200 KHz for semi-automated and automated collections.

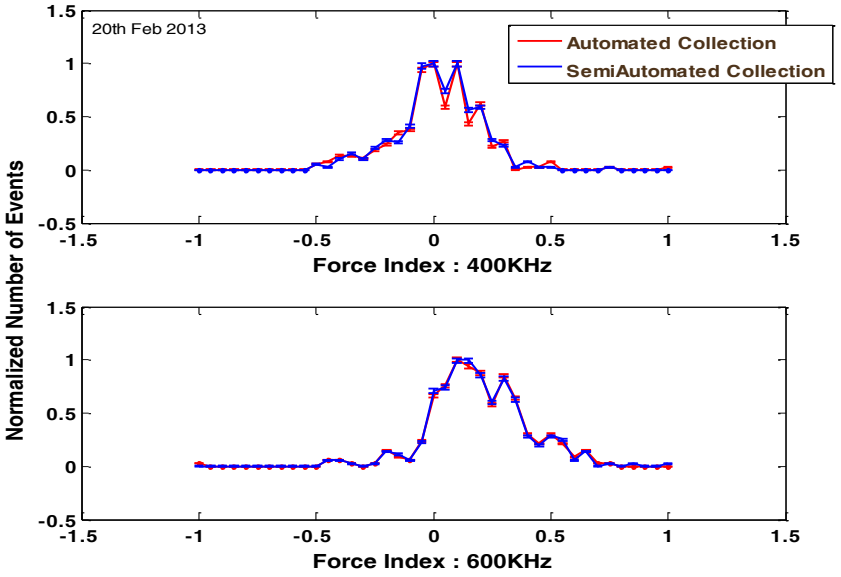


Figure 4.44: Uncertainty comparison in number of *DEP* events at 400 KHz and 600 KHz for semi-automated and automated collections.

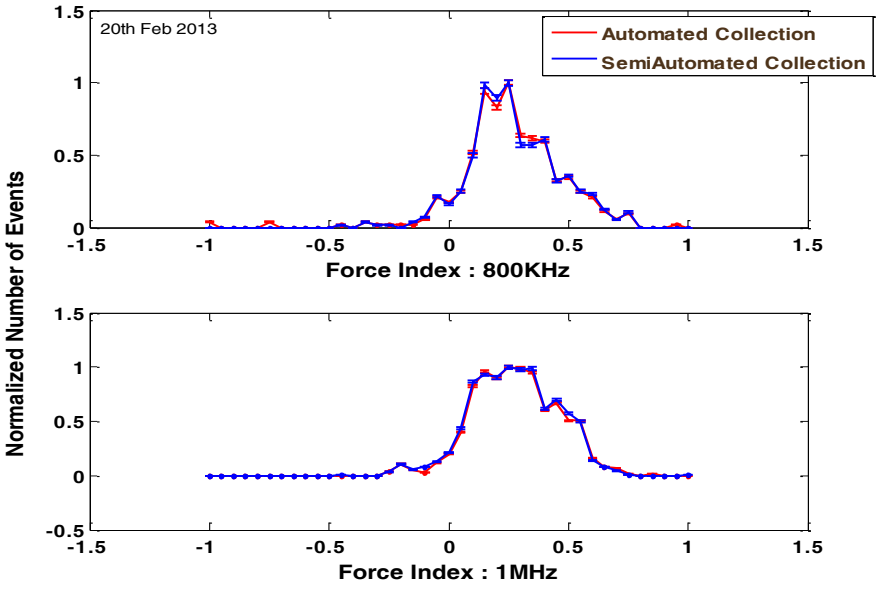


Figure 4.45: Uncertainty comparison in number of *DEP* events at 800 KHz and 1 MHz for semi-automated and automated collections

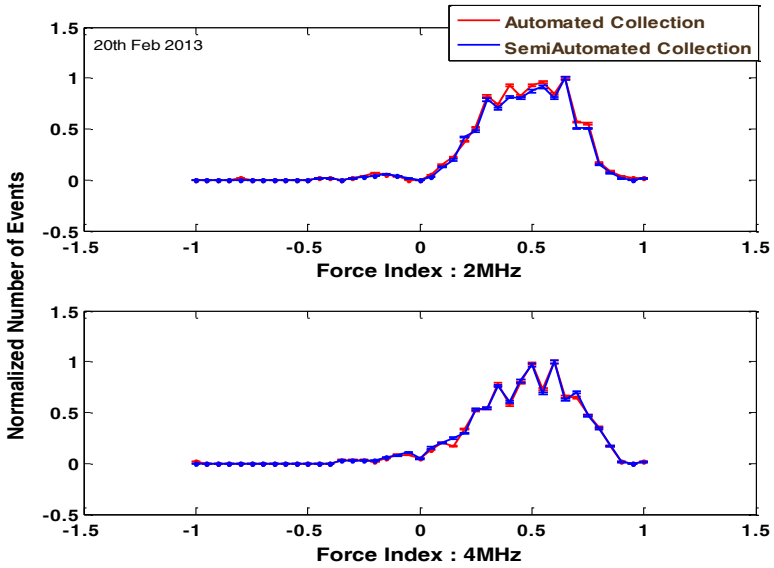


Figure 4.46: Uncertainty comparison in number of *DEP* events at 2 MHz and 4 MHz for semi-automated and automated collections.

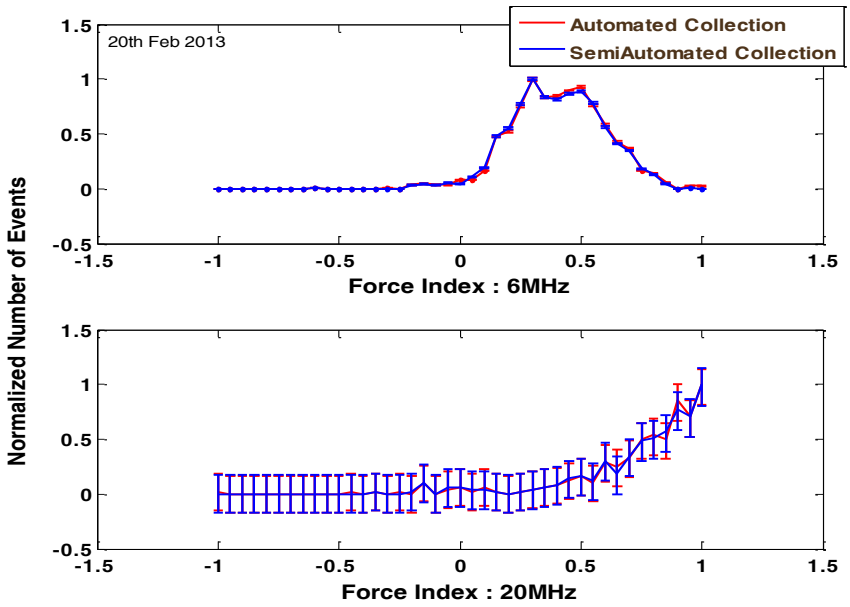


Figure 4.47: Uncertainty comparison in number of *DEP* events at 6 MHz and 20 MHz for semi-automated and automated collections.

Table 4.8 shows the comparisons of statistics for semi-automated and automated data collections at different frequencies. The average force index values and standard deviation of force index at all frequencies match. The percentage of signals lost due to automation is 8.9%. Percentage Viability calculation for both semi-automated and automated collections done at 6MHz seem to be similar. The uncertainties in percentage viabilities at 6MHz for both the data collection methods compared are ±3%.

Table 4.8: Comparison of statistics for semi-automated and automated data collections at different *DEP* frequencies. The bottom row of the table shows ranges of the force index between which semi-automated and automated data are compared for the *T*-tests.

Frequency	**Semi-Automated Collection**				**Automated Classification**				***T*-test**	
	N	**%Via.**	**μ_Φ**	**σ_Φ**	***N***	**%Via.**	**μ_Φ**	**σ_Φ**	***H***	**p-value**
50KHz	650	-	-0.29	0.09	588	-	-0.29	0.09	0	0.59
200KHz	421	-	-0.19	0.13	386	-	-0.18	0.22	0	0.32
400KHz	279	-	0.01	0.17	257	-	0.01	0.2	0	0.88
600KHz	294	-	0.19	0.2	277	-	0.19	0.2	0	0.30
800KHz	414	-	0.27	0.19	388	-	0.25	0.24	0	0.56
1MHz	793	-	0.28	0.18	688	-	0.27	0.17	0	0.64
2MHz	671	-	0.46	0.2	586	-	0.46	0.21	0	0.32
4MHz	614	-	0.45	0.22	562	-	0.46	0.24	0	0.39
6MHz	848	98.6±3%	0.4	0.19	783	97.8±3%	0.40	0.19	0	0.5
20MHz	287	-	0.76	0.28	287	-	0.75	0.31	0	0.63
Freq. (Hz)	50K	200K	400K	600K	800K	1M	2M	4M	6M	20M

Range of Φ	-0.55 <Φ < 0.05	-0.4<Φ< 0.1	-0.5< Φ< 0.5	-0.55 <Φ< 0.6	-0.1< Φ< 0.75	-0.2<Φ< 0.6	-0.05 <Φ< 0.75	-0.15 <Φ < 0.75	-0.1 <Φ < 0.8	-0.15< Φ< 0.8

The results of *T*-tests performed on data collected at different frequencies by both semi-automated and automated methods are also tabulated in Table 4.8. The limits of force index values between which semi-automated and automated data are compared in the *T*-test are also tabulated in Table 4.8. *H* and *p* are the hypothesis test results and the *p*-values respectively. *H=0* indicates that data sets tested in this statistical test are the same. All *p*-values are greater than 5% significance level indicating that the null hypothesis is true in all the cases.

4.9.3 Comparing Manual and Automated Methods

In this section, results of manual and automated data collection methods are compared for data obtained from electroporation studies. The dielectric response of a biological cell after being exposed to electroporating pulses is studied [43]. For electroporation studies, very low flow velocity is maintained to allow only one cell to enter the channel and it also gives time to switch the direction of flow in order to cycle the cell back and forth in the channel. For automated data collection, different window sizes are applied to different parts of the data separately in order to prevent loosing detection signals due to varying average velocities [43]. Segments of data analyzed with different window sizes are marked in Figure 4.34. The first *62* seconds of data is analyzed with a 300ms window; the next *205* seconds of data is analyzed with a 400ms window size, and the remaining *960* seconds of data is analyzed with a 250ms window size. In Figure 4.48, the change in

force index over time for manual and automated data collection methods is compared. Force index values obtained from both collections follow the same curve. In Figures 4.49 and 4.50, comparisons of peak 1 and peak 2 amplitudes over time for manual and automated data collection methods, respectively, are shown. The change in velocity over time is plotted in Figure 4.51 for both manual and automated data collection methods.

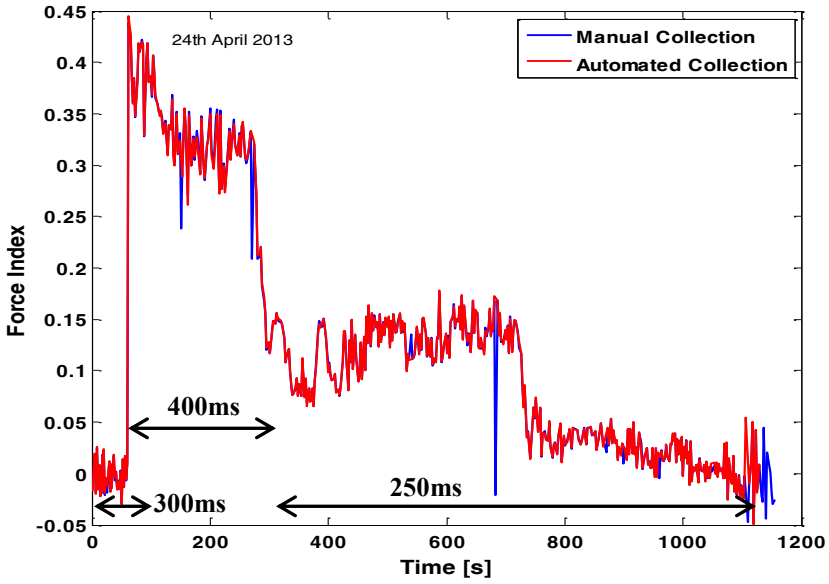


Figure 4.48: Comparison of change in force index over time for manual and automated collections. The segments of data collected with different window sizes are marked.

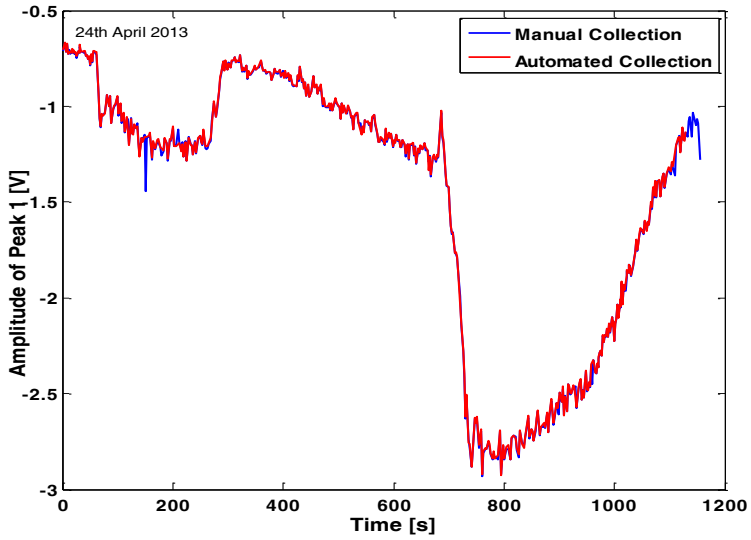


Figure 4.49: Comparison of change in amplitude of peak 1 over time for manual and automated collections.

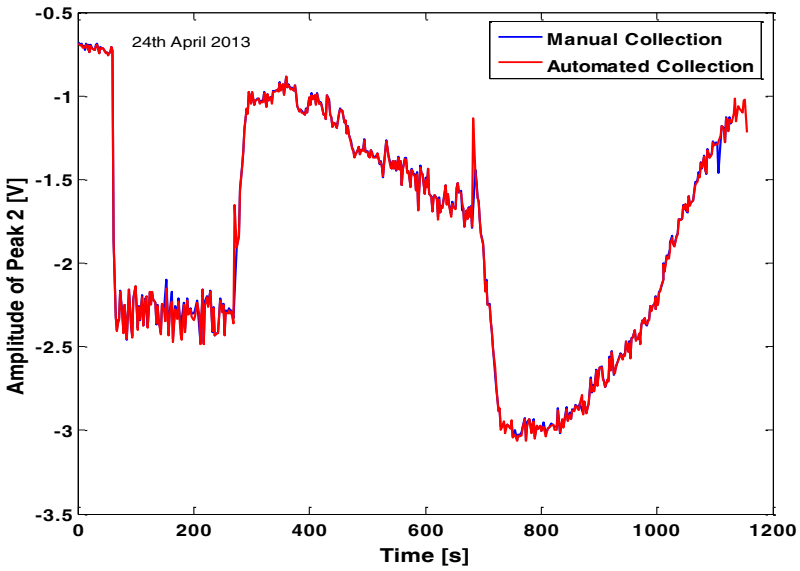


Figure 4.50: Comparison of change in amplitude of peak 2 over time for manual and automated collections.

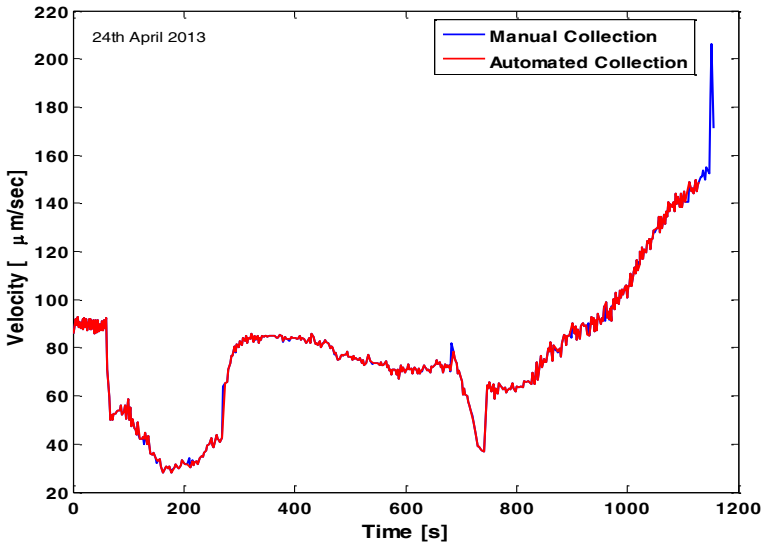


Figure 4.51: Comparison of change in average flow velocity over time for manual and automated collections.

4.9.4 Comparing Manual, Semi-Automated and Automated Methods

After validating semi-automated and automated data collection methods, it is important to compare them to the manual data collection. For comparison, detection signals collected from one experiment are compared between same time periods. The optimum window size for automated data collection is found to be 125ms. Figure 4.52 shows the comparison of force index distributions for manual, semi-automated and automated data collection methods. Uncertainties are calculated for force index bins with more than 10 events and are compared for all the data collections as shown in Figure 4.53. The uncertainties in the percentage viabilities for semi-automated and automated collections with respect to manual data collection are ±6%. Tables 4.9 and 4.10, compare

the statistics of control group and *DEP* force index distributions, respectively, for all types of data collection methods.

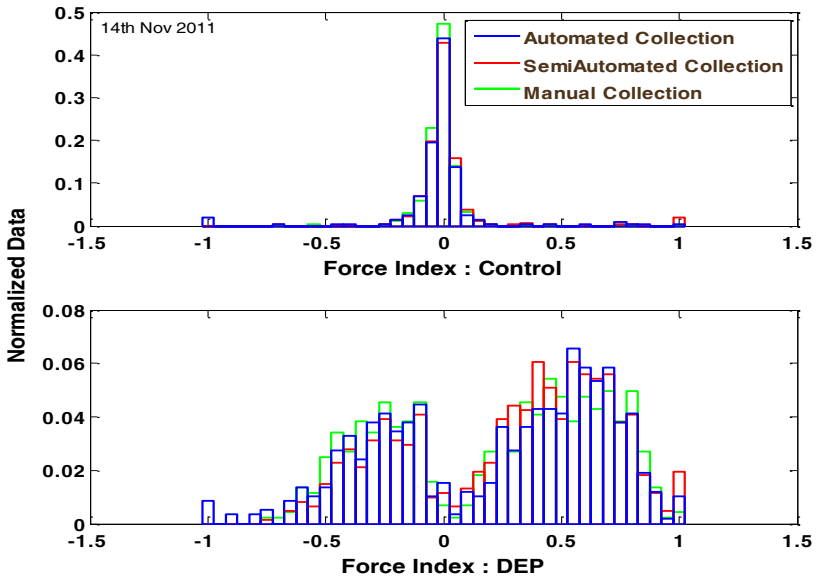


Figure 4.52: Comparison of force index distributions for manual, semi-automated and automated data collections.

Table 4.9: Comparison of control group statistics for manual, semi-automated and automated data collections.

Collection Method	**Statistics: Control Group**			
	N	**μ_Φ**	**σ_Φ**	**Loss %**
Manual	241	-0.014	0.066	-
Semi-Automated	215	-0.008	0.07	10%
Automated	192	-0.009	0.065	20%

Table 4.10: Comparison of *DEP* statistics for manual, semi-automated and automated data collections.

Collection Method	Statistics : *DEP*				
	N	**% Viability**	μ_Φ	σ_Φ	**Loss %**
Manual	442	62.2±6%	0.217	0.44	-
Semi-Automated	434	65.6±6%	0.293	0.46	1%
Automated	394	62.4±6%	0.227	0.45	10%

T-tests are performed on data collected by the three methods: Manual, Semi-Automated and Automated collections. Control group and *DEP* data for two types of collections are compared at a time. The hypothesis test results *H* and *p*-values for all the *T*-Tests are tabulated in Table 4.11. The *H*-values '0' indicate that null hypothesis is supported and the data sets compared are not significantly different.

Table 4.11: *T*-tests performed on data collected by all three methods: Manual, Semi-Automated and Automated. One *T*-test compares two data collections.

Data collection Methods Compared	***T*-Test**			
	H: Control Group	*p*-value: Control Group	*H*: *DEP*	*p*-value: *DEP*
Manual and Semi-Automated	0	0.35	0	0.32
Semi-Automated and Automated	0	0.95	0	0.17
Automated and Manual	0	0.37	0	0.75

The statistics of control group and *DEP* distributions suggest that mean force index values and standard deviations in the force index for all data collection methods are very close to each other. In automated collection, there is certainly some loss in the number of identified events because their identification is determined by the patterns used for ANN

training. Another reason is the fixed window size used throughout the entire event identification process.

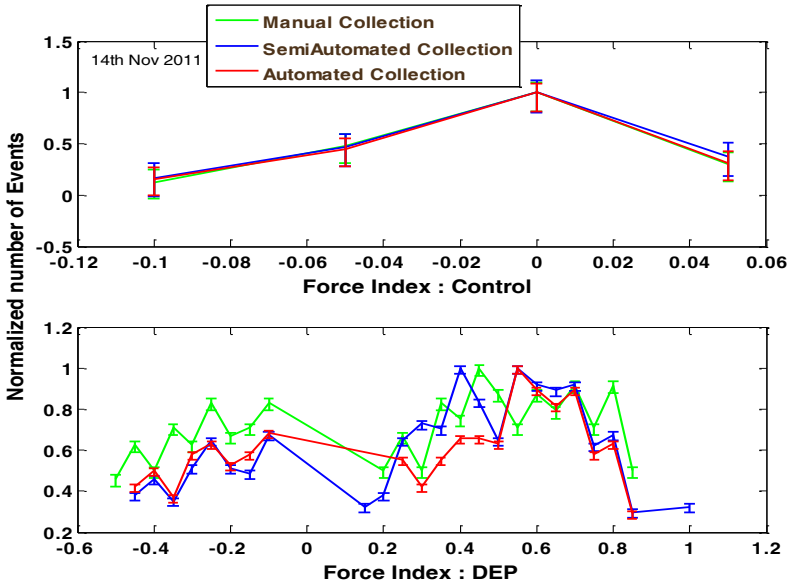


Figure 4.53: Uncertainty comparison in number of *DEP* events for semi-automated and automated collections.

For this particular experimental data, the semi-automated identifier and the automated identifier are capable of identifying 95% and 85% of the events identified manually respectively. For automated data collection, 10% loss in *DEP* events is acceptable as long as it does not skew the force index distributions generated from the manual and semi-automated data collections. The loss in control group events does not affect the percentage viability calculations and control group distributions are plotted just to ensure that the mean force index for unactuated cells lie at zero. The automated event classifier requires 10 seconds to analyze 100 detection signals when compared to the time required by the manual and semi-automated methods as tabulated in Table 4.12.

Table 4.12: Timed Collection of 100 detection signals by manual, semi-automated and automated data collection methods.

Data Collection Method	Collection Time for 100 signals (seconds)
Manual	476
Semi-Automated	162
Automated	10

The automated classifier reduces the signal classification time by 48 times when compared to classification time required by the manual method. The automated event classifier significantly reduces the signal analysis time, thus, allowing larger volumes of data to be analyzed in smaller time periods with an acceptable lower percentage of loss.

Chapter 5

Conclusion and Future Work

The aim achieved in this thesis is the development and design of signal processing methods to analyze detection signals post experiment. Two programs were written to analyze large volumes of data in a time period much lesser than the time required for manually analyzing detection signal collection.

The first program 'Semi-Automated Event Identifier', based on some features of the manual method, written to automatically identify detection signals as events is capable of identifying 87.8% of the signals collected by the manual method in a time period almost three times less than the time required by the manual method. The statistics obtained from the both the manual and the semi-automated methods closely match each other. The distributions obtained from the semi-automated method are validated by performing a T-test on analyzed data collected from both the manual and semi-automated methods and computing the uncertainties in the number of events. The uncertainties obtained from both the distributions overlap and the assumed null hypothesis of the T-test is supported.

The second program 'Automated Event Classifier' based on pattern recognition uses features of detection signals to classify them as accepted or rejected. An ANN was designed and trained to learn the patterns of detection signals. The automated event classifier significantly reduces the signal analysis time periods while efficiently classifying unseen detection signals. A low signal misclassification percentage of 3.1% is maintained. Automation does not in any way skew the statistics obtained from this

method when compared to the manual and semi-automated methods. Uncertainties and *T*-tests validate that the distributions obtained from all the analysis methods are indistinguishable.

Several data sets are analyzed and compared (only a few results are presented in this thesis) to arrive at the conclusion that the analysis techniques discussed in thesis are robust and efficient.

In future, the automated event classifier can be incorporated as a part of real-time processing. The automated event classifier could classify detection signals as and when they are acquired by the DAQ. This would not only eliminate the need for analysis post experiment but also allow miniaturization of the microfluidic device making it a portable, low-cost sensor for diagnostic purposes in healthcare.

References

[1] K. Pantel and R. H. Brackenhoff, "Dissecting the metastatic cascade.," *Nature Rev.,* vol. 4, p. 448, 2004.

[2] S. Suresh et al., "Connections between single-cell biomechanics and human disease states: gastrointestinal cancer and malaria.," *Acta Biomater.,* vol. 1, p. 15, 2005.

[3] H. A. Pohl, "The Motion and Precipitation of Suspensoids in Divergent Electric Fields," *J.Appl.Phys.,* vol. 22, p. 869, 1951.

[4] H. A. Pohl, Dielectrophoresis: The Behaviour of Neutral Matter in Nonuniform Electric Fields, Cambridge University Press, 1978.

[5] T. B. Jones, Electromechanics of Particles, Cambridge University Press, 1995.

[6] Z. Gagnon, "Cellular dielectrophoresis: applications to the characterization, manipulation, separation and patterning of cells.," *Electrophoresis,* vol. 32, p. 2466, 2011.

[7] R. Pethig, "Dielectrophoresis: Status of the theory, technology, and applications.," *Biomicrofluidics,* vol. 4, p. 022811, 2010.

[8] D. Holmes and H. Morgan, "Single Cell Impedance Cytometry for Identification and Counting of CD4 T-Cells in Human Blood Using Impedance Labels," *Analytical Chemistry,* vol. 82, no. 4, pp. 1455-1461, 2010.

[9] S. L. Stott et al., "Isolation of circulating tumor cells using a microvortex-generating herringbone-chip," *Proc. Natl. Acad. Sci. U.S.A.,* vol. 18392, p. 107, 2010.

[10] M. -T. Wei, J. Junio and H. D. Ou-Yang, "Direct measurements of the frequency-

dependent dielectrophoresis force," *Biomicrofluidics,* vol. 3, p. 012003, 2009.

[11] N. Demierre, R. Braschler, R. Muller and P. Renaud, "Focusing and continuous separation of cells in a microfluidic device using lateral dielectrophoresis.," *Sensors and Actuators B,* vol. 132, p. 388, 2008.

[12] Y. Huang, R. Holzel, R. Pethig and X. -B. Wang, "Differences in the AC electrodynamics of viable and non-viable yeast cells determined through combined dielectrophoresis and electrorotation studies.," *Phys.Med.Biol.,* vol. 37, no. 7, p. 1499, 1992.

[13] F. F. Becker et al., "Seperation of human breast cancer cells from blood by differential dielectric affinity," *Proc. Natl. Acad. Sci. U.S.A.,* vol. 92, p. 860, 1995.

[14] M. Toner and D. Irimia, "Blood-on-a-chip," *Annu. Rev. Biomed. Eng.,* vol. 7, p. 77, 2005.

[15] P. R. Gascoyne, X. -B. Wang, Y. Huang and F. F. Becker, "Dielectrophoretic separation of cancer cells," *IEEE Trans. Indus. Appl.,* vol. 33, p. 670, 1997.

[16] F. H. Labeed, H. M. Coley and M. P. Hughes, "Differences in the biophysical properties of membrane and cytoplasm of apoptotic cells revealed using dielectrophoresis," *Biochim Biophys Acta.,* vol. 922, p. 1760, 2006.

[17] L. A. Flangan et al., "Unique dielectric properties distinguish stem cells and their differentiated progeny," *Stem Cells,* vol. 26, no. 3, p. 656, 2008.

[18] P. R. Gascoyne, J. Noshari, T. J. Anderson and F. F. Becker, "Isolation of rare cells from cell mixtures by dielectrophoresis.," *Electrophoresis,* vol. 30, no. 8, p. 1388, 2009.

[19] S. A. Vanapalli, M. H. Duits and F. Mugele, "Microfluidics as a functional tool for cell mechanics," *Biomicrofluidics,* vol. 3, p. 012006, 2009.

[20] M. Nikolic-Jaric, S. F. Romanuik, G. A. Ferrier, T. Cabel, E. Salimi, D. B. Levin, G. E. Bridges and D. J. Thomson, "Electronic detection of dielectrophoretic forces exerted on particles flowing over interdigitated electrodes," *Biomicrofluidics,* vol. 6, no. 2, p. 024117, 2012.

[21] M. Nikolic-Jaric, T. Cabel, E. Salimi, A. Bhide, K. Braasch, M. Butler, G. E. Bridges and D. J. Thomson, "Differential electronic detector to monitor apoptosis using dielectrophoresis-induced translation of flowing cells (dielectrophoresis cytometry)," *Biomicrofluidics,* vol. 7, no. 2, p. 024101, 2013.

[22] M. Nikolic-Jaric, S. F. Romanuik, G. A. Ferrier, G. E. Bridges, M. Butler, K. Sunley and D. J. Thomson, "Microwave frequency sensor for detection of biological cells in microfluidic channels," *Biomicrofluidics,* vol. 3, no. 3, p. 034103, 2009.

[23] F. M. Hughes, C. D. Bortner, G. D. Purdy and J. A. Cidlowski, "Intracellular K+ Suppresses the Activation of Apoptosis in Lymphocytes," *Bio.Chem.,* vol. 272, no. 48, pp. 30567-30576, 1997.

[24] M. Nikolic-Jaric, E. Salimi, T. Cabel, K. Braasch, M. Butler, G. E. Bridges and D. J. Thomson, "Dielectrophoretic (*DEP*) cytometer: Label-free electronics sensing of physiological changes in cells," in *6th International Conference on Miniaturized Systems for Chemistry and Life Sciences*, Okinawa,Japan, 2012.

[25] G. H. Markx and C. L. Davey, "The dielectric properties of biological cells at radiofrequencies: applications in biotechnology," *Enzyme and Microbial*

Technology, vol. 25, no. 8, pp. 161-171, 1999.

[26] H. Morgan, S. Tao, D. Holmes, S. Gawad and N. G. Green, "Single cell dielectric spectroscopy," *J. Phys. D: Appl. Phys.,* vol. 40, p. 61, 2007.

[27] J. C. Maxwell, A Treatise on Electricity and Magnetism, New York: Dover, 1954.

[28] A. Valero, T. Braschler and P. Renaud, "A unified approach to dielectric single cell analysis: Impedance and dielectrophoretic force spectroscopy," *Lab on a Chip,* vol. 10, pp. 2216-2225, 2010.

[29] M. Schmeer, T. Seipp, U. Pliquett, S. Kakorin and E. Neumann, "Mechanism for the conductivity changes caused by membrane electroporation of CHO cell-pellets," *Phys. Chem. Chem. Phys.,* vol. 6, pp. 5564-5574, 2004.

[30] H. Saito, Y. Suzuki and M. Taki, "Measurement of complex permittivity for biological cells at 1.7-2.6 GHz by waveguide penetration method," *URSI General Assembly, Paper K01p2,* 2008.

[31] T. I. Cabel, "Dielectrophoretic Analysis of Polystyrene Spheres in Fluidic Suspension," University of Manitoba, Winnipeg, 2013.

[32] S. F. Romanuik, "A microflow cytometer with simultaneous dielectrophoretic actuation for the optical assay and capacitive cytometry of individual fluid suspended bioparticles," University of Manitoba, Winnipeg, 2009.

[33] E. Billauer, *peakdet: Peak detection using matlab.[Online].Available: http://billauer.co.il/peakdet.html,* 2012,July.

[34] B. S. Rizi, A. Bhide, T. Cabel, M. Nikolic-Jaric, E. Salimi, K. Braasch, M. Butler, G. E. Bridges and D. J. Thomson, "Semi-automated detection of single cell signatures

from a dielectrophoretic cytometer," in *IEEE International Instrumentation and Measurement Technology Conference*, Minneapolis,MN, 2013.

[35] Wikipedia, "Student's t-test," [Online]. Available: http://en.wikipedia.org/wiki/Student%27s_t-test. [Accessed 4th December 2013].

[36] "Bench Prep," [Online]. Available: https://benchprep.com/blog/ap-biology-animal-systems-vi-nervous-system-part-1/. [Accessed 20th October 2013].

[37] Wikipedia, "Neuron," [Online]. Available: http://en.wikipedia.org/wiki/Neuron. [Accessed 20th October 2013].

[38] M. T. Hagan, H. B. Demuth and M. Beale, Neural Network Toolbox™ Getting Started Guide, The MathWorks, Inc., 1992-2013.

[39] Z. Yu, "Feed-forward Neural Networks and Their Applications in Forecasting," University of Houston, Texas, 2000.

[40] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning representations by back-propagating errors," *Nature,* vol. 323, pp. 533-536, 1986.

[41] M. F. MØller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks,* vol. 6, no. 4, pp. 525-533, 1993.

[42] S. Brueckner, *crossing: Detect zero (or any other) level crossing of signals.[Online].Available: http://www.mathworks.com/matlabcentral/fileexchange/2432-crossing,* 2002, September 25th.

[43] E. Salimi, M. Nikolic-Jaric, T. Cabel, K. Braasch, V. Jung, M. Butler, D. J. Thomson and G. E. Bridges, "Microfluidic device for simultaneous pulsed electric

field electroporation and dielectrophoresis studies of single biological cells," in *IEEE Instrumentation and Measurement Technology Conference* , Minneapolis,MN, 2013.

[44] G. A. Ferrier, "Electrical detection and actuation of single with application to deformability cytometry for markerless diagnostics," University of Manitoba, Winnipeg, 2012.

[45] P. Gascoyne and J. V. Vykoukal, "Dielectrophoresis-based sample handling in general-purpose programmable diagnostic instruments," *Proceedings of IEEE Inst Electr Electron Eng,* vol. 92, no. 1, pp. 22-42, 2004.

[46] G. A. Ferrier, S. F. Romanuik, D. J. Thomson, G. E. Bridges and M. R. Freeman, "A microwave interferometric system for simultaneous actuation and detection of single biological cells," *Lab on Chip,* vol. 9, no. 23, pp. 3406-3412, 2009.