Histogram clustering for rapid time-domain fluorescence lifetime image analysis

YAHUI LI,1,2,6 NATAKORN SAPERMSAP,3 JUN YU,4 JINSHOU TIAN,1,2 YU CHEN,3 AND DAVID DAY-UEI LI5,7

1Key Laboratory of Ultra-fast Photoelectric Diagnostics Technology, Xi’an Institute of Optics and Precision Mechanics, Xi’an Shaanxi 710049, China
2Collaborative Innovation Center of Extreme Optics, Shanxi University, Taiyuan Shanxi 030006, China
3Department of Physics, Scottish Universities Physics Alliance, University of Strathclyde, Glasgow, G4 0NG, United Kingdom
4Strathclyde Institute of Pharmacy and Biomedical Sciences, University of Strathclyde, Glasgow G4 0RE, United Kingdom
5Department of Biomedical Engineering, University of Strathclyde, Glasgow G1 0NW, United Kingdom
6yahuili777@hotmail.com
7David.Li@strath.ac.uk

Abstract: We propose a histogram clustering (HC) method to accelerate fluorescence lifetime imaging (FLIM) analysis in pixel-wise and global fitting modes. The proposed method’s principle was demonstrated, and the combinations of HC with traditional FLIM analysis were explained. We assessed HC methods with both simulated and experimental datasets. The results reveal that HC not only increases analysis speed (up to 106 times) but also enhances lifetime estimation accuracy. Fast lifetime analysis strategies were suggested with execution times around or below 30 µs per histogram on MATLAB R2016a, 64-bit with the Intel Celeron CPU (2950M @ 2GHz).

Published by The Optical Society under the terms of the Creative Commons Attribution 4.0 License. Further distribution of this work must maintain attribution to the author(s) and the published article’s title, journal citation, and DOI.

1. Introduction

Fluorescence lifetime imaging (FLIM) [1] is a crucial technique for assessing microenvironments of fluorophores, such as pH, Ca$^{2+}$, O$_2$, viscosity, or temperature [2–5]. Combining with Förster Resonance Energy Transfer (FRET) techniques [6], FLIM can be a powerful “quantum ruler” to measure protein conformations and interactions [7]. In contrast to fluorescence intensity imaging, FLIM is independent of fluorescence intensities and fluorophore concentrations, making FLIM a robust quantitative imaging technique for life sciences applications [8,9], medical diagnosis [10], drug developments [11,12], and flow diagnosis [13–15].

A fluorescence decay is usually modeled as a sum of exponential decay functions:

\[ f(t) = A \sum_{p=1}^{P} q_p \exp(-t/\tau_p), \sum_{p=1}^{P} q_p = 1, \] (1)

where $A$ is the amplitude, $q_p$ and $\tau_p$ are the fraction and lifetime of the $p^{th}$ component, $p = 1, \ldots, P$. In vector forms, $q = [q_1, \ldots, q_P]^T$ and $\tau = [\tau_1, \ldots, \tau_P]^T$. In reality, the measured signal is a convolution of $f(t)$ and the instrument response function (IRF) $i(t)$,

\[ h(t) = if(t) * f(t) + \epsilon(t), \] (2)

where $\epsilon(t)$ is noise.

FLIM analysis is equivalent to solving the inverse problem from Eq. (2) with the measured $h(t)$ to obtain $q$ and $\tau$. FLIM experiments can be conducted either in time- or frequency-domain.
manners [8]. In time-domain approaches, samples are illuminated with ultrashort laser pulses. 

\( h(t) \) is measured using a time-correlated single-photon counting (TCSPC) system [16,17] with photomultiplier tubes, delay line anode detectors [18] or single-photon avalanche diodes [19] in scanning or widefield modes. \( h(t) \) can also be measured with time-gated cameras [20,21] and streak cameras [22,23]. There are also frequency-domain approaches [24,25], but we will focus on time-domain approaches in this report.

A fluorescence decay histogram measured by a TCSPC system can be:

\[
    h_m = \sum_{k=0}^{m} if_{k-m} \cdot f_m + \epsilon_m, \tag{3}
\]

where

\[
    if_m = \int_{m\Delta t}^{(m+1)\Delta t} if(t)dt, f_m = \int_{m\Delta t}^{(m+1)\Delta t} f(t)dt, m = 0, \ldots, M - 1. \tag{4}
\]

\( M \) is the number of time-bins, and \( \Delta t \) is the bin width (the TCSPC’s temporal resolution). We can express Eq. (3) in a vector form with

\[
    h = [h_0, \ldots, h_{M-1}]^T, \quad if = [if_0, \ldots, if_{M-1}]^T, \quad f = [f_0, \ldots, f_{M-1}]^T.
\]

With \( h \) and \( if \) already measured, \( A, q \) and \( \tau \) can be extracted with a lifetime determination algorithm by solving a nonlinear minimization problem \( \arg \min ||h - \hat{h}||^2 \), where \( \hat{h} \) is the estimated histogram. The iterative convolution (IC) is commonly used with the least-squared method (LSM) [26,27] for solving the inverse problem, denoted as IC-LSM. Still, IC-LSM suffers from low photon efficiency and slow analysis. Several deconvolution approaches have been developed to enhance the analysis, such as the Laguerre expansion [28–30], the non-fitting and the global fitting [31,32] methods. The Laguerre expansion methods speed up deconvolution procedures by converting the nonlinear-fitting problem to a linear-fitting problem estimating a Laguerre basis set’s expansion coefficients. The non-fitting methods, including the centre-of-mass method (CMM) [33–35], the integral extraction method (IEM) [36,37], the phasor method [38,39], or the rapid lifetime determination method [40,41], can provide rapid average lifetime analysis [42]. The global fitting methods can accelerate analysis by changing the estimation mode from the pixel-wise mode to a global fitting mode and using spatial lifetime invariances of fluorescent species in an image to reduce the degree of freedom significantly. There are two strategies, IC [31] and the variable projection (VP) method [32], for implementing global fitting.

However, the Laguerre expansion, the non-fitting and the global fitting methods are not fast enough for growing demands for real-time FLIM. This work presents a histogram clustering (HC) method for improving FLIM analysis in analysis speed and accuracy. Section 2 (Methods) summarizes the workflows for decay parameter image reconstructions with and without HC. We will then introduce and demonstrate the HC method. Besides the algorithms used in this work, HC can also accelerate other algorithms, such as the maximum likelihood method [43], Bayesian methods [44], and deep-learning methods [45]. In Section 3 (Results and Analysis), synthetic and experimental TCSPC datasets will be used to evaluate the HC method’s performances. Suggestions of the fastest algorithms for different outcomes will be given.

2. Methods

2.1. Modes for decay parameter image reconstructions

Figure 1(a) shows the Pixel-Wise (PW) mode’s workflow. \( N_v \) is the number of valid pixels in a TCSPC dataset whose intensities are beyond a threshold. Histogram \( s \), denoted as \( h^{(s)} \), is sent into an algorithm for PW along with \( if_s \), \( s = 1, \ldots, N_v \). After \( N_v \) histograms are analyzed pixel-by-pixel, decay parameter images are produced. The total execution time \( t_{exe}^{PW} = N_v \times t_A^{PW} \), \( t_A^{PW} \) is the adopted algorithm’s execution time for PW.
Figure 1(b) shows the workflow of the Global Fitting mode for all Pixels, denoted as GF-P. Instead of estimating decay parameters individually for each pixel, GF-P assumes lifetimes $\tau$ are constants, and $A$ and $q$ vary across the image. $t_{exe}^{GF-P} = t_A^{GF}$. $t_A^{GF}$ is the adopted algorithm’s execution time for GF.

Figure 2 shows the workflows where the HC method is embedded. Figure 2(a) shows the Cluster-Wise (CW) mode, which combines PW and HC; likewise, Fig. 2(b) shows the Global Fitting mode for all Clusters (GF-C), which combines GF-P and HC.

In CW, $N_p$ histograms are first sorted by HC, whose execution time is $t_{HC}$. into $N_c$ classes with $N_c$ cluster-histograms $\hat{h}^{(s)}$, $s = 1, \ldots, N_c$. $\hat{h}^{(s)}$ is used to estimate decay parameters for Cluster $s$. Then, the decay parameters are assigned to the corresponding cluster’s pixels with a parameter assignment function, whose execution time is $t_{PA}$. Therefore, $t_{exe}^{CW} = t_{HC} + N_c \times t_{PW} + t_{PA}$.

In GF-C, $N_p$ histograms are processed with HC first, and the output $N_c$ histograms are sent into an algorithm for GF. Decay parameters for all clusters are obtained and assigned to the pixels in corresponding clusters with the parameter assignment function. Therefore, $t_{exe}^{GF-C} = t_{HC} + t_A^{GF} + t_{PA}$. 
The algorithms used in this work are reviewed in Supplement 1.

2.2. Histogram clustering

In reality, there are always many pixels within the field of view showing similar histogram profiles, and it’s unnecessary to analyze them individually because it would be time-consuming. The idea of HC is to sort histograms with similar profiles and to divide them into \( N_c \) clusters. If histograms have similar decay profiles, they are supposed to show similar decay parameters. Therefore, we can average similar decay profiles in a cluster into one profile to estimate decay parameters and then assign them to all pixels. With this arrangement, we only need to process \( N_c \) instead of \( N_v \) histograms. HC significantly speeds up FLIM analysis.

For simplicity, we only discuss bi-exponential decays widely used in practice. Figure 3(a) shows an IRF and normalized signal profiles \( h(t) \) following a bi-exponential decay model, and Fig. 3(b) shows corresponding cumulative signals, \( H(t) = \int_0^t h(t)dt \), which is not sensitive to Poisson noise due to the integration. Signal decay parameters are also labelled in Fig. 3(a). If we choose an intensity bound, \( I_{\text{bound}} \), then each signal has a corresponding time delay, \( t_{I_b} \), to reach \( I_{\text{bound}} \), as shown in Fig. 3(b).

![Fig. 3. Illustrations of (a) an IRF and normalized signal profiles \( h(t) \) following a bi-exponential decay model and (b) cumulative signal profiles \( H(t) \).](image)

It is straightforward for mono-exponential decays, \( f(t) = A \exp(-t/\tau) \), that \( t_{I_b} \) has an approximately linear relationship with \( \tau \). Figure 4 shows \( t_{I_b} \) curves with different IRFs which introduce time-shifts (assuming that IRFs for all histograms are the same in a scanning system). If a multichannel sensor is used, IRF alignments are required before using HC.

![Fig. 4. \( t_{I_b} \) of signals following mono-exponential models depending on \( \tau \) under different IRFs.](image)

However, it is less straightforward for bi-exponential decays. Thus, we used numerical methods to conduct analysis, as shown in Fig. 5, in which three cases were simulated to explain how the
proposed concept works. Case A has $q_1 = 0.5$, $0.1 \leq \tau_1 \leq 3$ ns, and $\tau_2 = 3$ ns as shown in Fig. 5(a); Case B has $0 \leq q_1 \leq 1$, $\tau_1 = 0.5$ ns, and $\tau_2 = 3$ ns as shown in Fig. 5(b); Case C has $q_1 = 0.5$, $\tau_1 = 0.5$ ns, and $0.5 \leq \tau_2 \leq 10$ ns as shown in Fig. 5(c). The IRF follows a Gaussian distribution with an FWHM of 0.5 ns.

Fig. 5. $t_b$ for (a) Case A: $q_1 = 0.5$, $\tau_1 = 0.1 \sim 3$ ns, and $\tau_2 = 3$ ns, (b) Case B: $q_1 = 0 \sim 1$, $\tau_1 = 0.5$ ns, and $\tau_2 = 3$ ns, (c) Case C: $q_1 = 0.5$, $\tau_1 = 0.5$ ns, and $0.5 \leq \tau_2 \leq 10$ ns with different $I_{bound}$.

For Case A, $t_b$ is not monotonic with $\tau_1$, and the monotonic range and the slope are functions of $I_{bound}$. As $I_{bound}$ increases, the slope increases with a smaller monotonic range. The profiles with $\tau_1$ outside the range are wrongly sorted into a cluster with a larger $\tau_1$. The monotonic ranges for $I_{bound} = 0.2$ and 0.6 are $0.4 \sim 3$ ns and $1 \sim 3$ ns, respectively. For Cases B and C, $t_b$ is monotonic (decreasing and increasing) with $q_1$ and $\tau_2$ for all $I_{bound}$, respectively. For the signals like Cases A $\sim$ C (which only have one variable), we can cluster the signals by $t_b$ with a proper $I_{bound}$ considering the monotonic range. For example, for Case A, if the shortest lifetime is around 0.5 ns, $I_{bound} = 0.2$ is a proper choice; for Cases B and C, $I_{bound}$ can be set arbitrarily in $0.1 \sim 1$. We use $I_{bound} = 0.2$ hereafter.

However, it is not realistic that the signals in a dataset have one variable and two constant decay parameters. For instance, in FRET-FLIM applications, donors without FRET have a constant lifetime and donors interacting with acceptors have shorter lifetimes due to FRET. Therefore, the short and long lifetimes, $\tau_1$ and $\tau_2$, are donors’ lifetimes with and without FRET, and $q_1$ is the portion of the donors undergoing FRET among all donors. $q_1$ and $\tau_1$ are variables depending on FRET efficiency.

For FRET-FLIM datasets, such as Case D: $q_1 = 0 \sim 1$, $\tau_1 = 0.5 \sim 3$ ns, and $\tau_2 = 3$ ns, with two variables, it is not enough to divide the histograms only depending on $t_b$, as histograms with different profiles would have the same $t_b$ and be wrongly divided into one cluster. Figure 6(a) shows the resulting clusters ($N_1 = 15$) in different colors for Case D with $M = 256$ depending on $t_b$. Figures 6(b) and (c) show the cumulative signals in Clusters 14 and 7 in red, respectively, and the averaged cumulative histograms $\bar{H}$ for the clusters (green dash lines). When $q_1 \sim 0$ (or $q_1 \sim 1$) or $\tau_1 \sim \tau_2$ (such as Clusters 14 and 15), the signals are nearly mono-exponential and have similar profiles, as shown in Fig. 6(b). However, the signals for other clusters (for example, Cluster 7) have the same $t_b$, but the profiles after $t_b$ diverge. At $t = I_{bound}$ in Fig. 6(c), the cumulative intensity $I_b$ in Cluster 7 is within [0.7, 0.9]. Therefore, we can further divide each cluster into $N_2$ sub-clusters depending on $I_b$.

Setting a larger $N_2$ can result in a higher clustering precision, which means histogram profiles in one cluster are more similar. Another way to increase clustering precision is interpolating $h$ with $M$ time-bins to $h^{interp}$ with $M \cdot N_{interp}$ time-bins. $N_{interp}$ ($\geq 1$) is an interpolation factor, and $h^{interp}$ can be expressed as

$$h^{interp}_{mN_{interp}+n} = h_m/N_{interp}, n = 0, \ldots, N_{interp} - 1, m = 0, \ldots, M - 1.$$  (5)
Fig. 6. For Case D with \( M = 256 \), (a) clusters, cumulative signals (red solid lines) and averaged cumulative signals (green dash lines) for (b) Cluster 14 and (c) Cluster 7.

Figure 7(a) shows the clusters for Case A with \( M = 256, N_{\text{interp}} = 1 \), and \( N_1 = 6 \). Figure 7(b) shows the clusters for \( M = 256, N_{\text{interp}} = 2 \), and \( N_1 = 11 \). The histograms in each cluster have a smaller range of \( \tau_1 \), leading to a higher clustering precision.

Fig. 7. For Case A, clusters with (a) \( M = 256 \) and \( N_{\text{interp}} = 1 \) and (b) \( M = 256 \) and \( N_{\text{interp}} = 2 \).

The HC workflow is summarized in Fig. 8(a). There are three steps: 1) depending on \( I_b \), \( N_v \) histograms are divided into \( N_1 \) clusters, which can be adjusted by setting \( N_{\text{interp}} \); 2) depending on \( I_{vb} \), histograms in each of the \( N_1 \) clusters are further divided into \( N_2 \) sub-clusters, and \( N_c = N_1 \times N_2 \) clusters are finally produced; 3) \( N_c \) histograms are generated by obtaining the averaged histogram in each cluster.

Fig. 8. (a) Workflow of the HC method. (b) Boxplot of \( \chi^2 \) for Case D with different \( N_{\text{interp}} \) and \( N_2 \).
To assess HC in terms of $N_{\text{interp}}$ and $N_2$, we can define:

$$
\chi^2(s) = \frac{1}{M} \sum_{m=0}^{M-1} \left| h_m^{(s)} - \bar{h}_m^{(s)} \right| / h_m^{(s)},
$$

(6)

where $h_m^{(s)}$ is the cluster histogram produced with HC for the cluster including Pixel $s$. Figure 8(b) shows the boxplot of $\chi^2(s)$ for Case D for various combinations of $N_{\text{interp}}$ and $N_2$. Poisson noise is included in each signal with a total intensity of $I = 10^4$ photon counts. The higher $N_{\text{interp}}$ and $N_2$ are, the lower $\chi^2(s)$ becomes, meaning higher accuracy. We set $N_{\text{interp}} = 2$ and $N_2 = 4$ for HC used in CW and GF-C modes in this work.

3. Results and analysis

Synthetic and experimental TCSPC datasets were used to assess the performances of HC. Table 1 summarizes different output types of algorithms: (1) the fitting method LE-LSM in PW and CW modes can produce $q$, $\tau$, $\tau_A$, and $\tau_I$ images; (2) the non-fitting methods LE-IEM and CMM in PW and CW modes can produce $\tau_A$ and $\tau_I$ images; (3) the global fitting methods IC and VP in GF-P and GF-C modes can produce $q$, $\tau_A$, and $\tau_I$ images and constant $\tau$. $\tau_A$ and $\tau_I$ are two types of average lifetimes, amplitude- and intensity-weighted average lifetimes,

$$
\tau_A = \sum_{p=1}^{P} q_p \tau_p, \quad \tau_I = \sum_{p=1}^{P} q_p \tau_p^2 / \sum_{p=1}^{P} q_p, \quad (7)
$$

Using IEM to calculate $\tau_A$ requires conducting the deconvolution first. The Laguerre expansion method is employed for deconvolution when IEM is used; therefore, we denote the whole process as LE-IEM.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Algorithm</th>
<th>Output variables$^a$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pixel-Wise (PW)/Cluster-Wise (CW)</td>
<td>Least-Squared Method with Laguerre Expansion LE-LSM</td>
<td>$I$ $I$ $I$ $I$</td>
</tr>
<tr>
<td></td>
<td>Integral Extraction Method with Laguerre Expansion LE-IEM</td>
<td>$X$ $X$ $X$ $I$</td>
</tr>
<tr>
<td></td>
<td>Centre-of-Mass Method CMM</td>
<td>$X$ $X$ $I$ $X$</td>
</tr>
<tr>
<td>Global-Fitting for all Pixels (GF-P)/Global-Fitting for all Clusters (GF-C)</td>
<td>Iterative Convolution IC</td>
<td>$I$ $C$ $I$ $I$</td>
</tr>
<tr>
<td></td>
<td>Variable Projection VP</td>
<td>$I$ $C$ $I$ $I$</td>
</tr>
</tbody>
</table>

$^a$Letters $I$ and $C$ represent that the outputs are images and constants, respectively. Letter $X$ stands for no output.

3.1. Simulated data

The synthetic TCSPC dataset has an image size of $150 \times 150$ pixels and $M = 256$. The simulated signals are bi-exponential ($P = 2$). Figure 9(a) shows the $\log_{10}(I_i)$ image consisting of three regions with integrated intensities of $I_1 = 500$, $I_2 = 1000$, and $I_3 = 10000$, respectively. Poisson noise is included in the dataset. Figures 9(b), 9(c), and 9(d) shows the $q_1$, $\tau_1$, and $\tau_2$ images, respectively. The $q_1$ image has three regions with mean values of [0.2, 0.5, 0.8] and relative standard deviations of 10%; the $\tau_1$ image has two regions with mean values of [0.5, 1] ns and
relative standard deviations of 10%; the \( \tau_2 \) image has a mean value of 3 ns and a relative standard deviation of 10%.

The execution times (\( t_{exe} \)) and the mean squared errors (MSE) of the results evaluated by different algorithms without and with HC are summarized in Table 2. MSE is defined as:

\[
MSE = \frac{1}{N_p} \sum_{s=1}^{N_p} |a^{(s)} - \hat{a}^{(s)}|^2,
\]

where \( \hat{a} \) represents the estimated \( a (a = q_1, \tau_1, \tau_2, \tau_A, \tau_I) \). The results for the fitting and non-fitting methods in PW and CW modes and the global fitting methods in GF-P and GF-C modes are illustrated and analyzed in the following sections.

### Table 2. \( t_{exe} \) and MSE evaluated by algorithms without and with HC.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Algorithm</th>
<th>( t_{exe} ) (s)</th>
<th>MSE ( q_1 )</th>
<th>MSE ( \tau_1 ) (ns(^2))</th>
<th>MSE ( \tau_2 ) (ns(^2))</th>
<th>MSE ( \tau_A ) (ns(^2))</th>
<th>MSE ( \tau_I ) (ns(^2))</th>
</tr>
</thead>
<tbody>
<tr>
<td>PW</td>
<td>LE-LSM</td>
<td>389.45</td>
<td>0.019</td>
<td>0.173</td>
<td>0.198</td>
<td>0.110</td>
<td>0.027</td>
</tr>
<tr>
<td></td>
<td>LE-IEM</td>
<td>62.30</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>CMM</td>
<td>0.20</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>0.185</td>
</tr>
<tr>
<td></td>
<td>IC</td>
<td>724.82</td>
<td>0.100</td>
<td>X</td>
<td>X</td>
<td>0.678</td>
<td>1.098</td>
</tr>
<tr>
<td></td>
<td>VP</td>
<td>3.34</td>
<td>0.033</td>
<td>X</td>
<td>X</td>
<td>0.122</td>
<td>0.178</td>
</tr>
<tr>
<td>GF-P</td>
<td>IC</td>
<td>11.85</td>
<td>0.017</td>
<td>X</td>
<td>X</td>
<td>0.102</td>
<td>0.050</td>
</tr>
<tr>
<td></td>
<td>VP</td>
<td>0.31</td>
<td>0.014</td>
<td>X</td>
<td>X</td>
<td>0.048</td>
<td>0.093</td>
</tr>
</tbody>
</table>

#### 3.1.1. Fitting method LE-LSM in PW and CW modes

Figure 10 shows \( \hat{q}_1, \hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_A, \) and \( \hat{\tau}_I \) images estimated with LE-LSM in (a1) – (a5) PW (without HC) and (b1) – (b5) CW (with HC) modes, respectively. The pixel brightness of each image represents the intensity. The parameter histograms (\( \hat{q}_1, \hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_A, \) and \( \hat{\tau}_I \)) of different intensity regions (blue, red, and magenta lines for \( I_1, I_2, \) and \( I_3 \) respectively) and the true histogram (black dash line) are attached to each image. HC improves the estimated images, especially for Regions \( I_1 \) and \( I_2 \), as the histograms are closer to the truth than those estimated without HC. MSEs are reduced from 0.019 to 0.011 for \( q_1 \), from 0.173 ns\(^2\) to 0.102 ns\(^2\) for \( \tau_1 \), from 0.198 ns\(^2\) to 0.104 ns\(^2\) for \( \tau_2 \), and from 0.110 ns\(^2\) to 0.027 ns\(^2\) for the global fitting methods in GF-P and GF-C modes.
ns² for \( \tau_2 \), from 0.110 ns² to 0.037 ns² for \( \tau_A \), and from 0.027 ns² to 0.025 ns² for \( \tau_I \). \( t_{exe} \) is significantly reduced from 389.45 s to 3.36 s.

![Fig. 10. \( \hat{q}_1 \), \( \hat{\tau}_1 \), \( \hat{\tau}_2 \), \( \hat{\tau}_A \), and \( \hat{\tau}_I \) images with LE-LSM in (a1) – (a5) PW (without HC) and (b1) – (b5) CW (with HC) modes. Histograms of different intensity regions (blue, red, and magenta for \( I_1 \), \( I_2 \), and \( I_3 \), respectively) and the true histogram (black dash line).](image)

### 3.1.2. Non-fitting methods LE-IEM and CMM in PW and CW modes

Figure 11 shows \( \hat{\tau}_A \) and \( \hat{\tau}_I \) images and histograms produced by LE-IEM and CMM (a) – (b) without and (c) – (d) with HC. LE-IEM is for estimating \( \hat{\tau}_A \). \( MSE(\tau_A) \) is improved from 0.102 ns² to 0.038 ns², and \( t_{exe} \) is reduced from 62.30 s to 0.63 s with HC.

![Fig. 11. \( \hat{\tau}_A \) and \( \hat{\tau}_I \) images and histograms produced by LE-IEM and CMM (a) – (b) without and (c) – (d) with HC.](image)

CMM is for estimating \( \hat{\tau}_I \) with the shortest \( t_{exe} \) either in PW or CW, around 0.20 s, but it has a bias, as shown in Figs. 11(b) and 11(d). \( MSE(\tau_I) \) is around 0.180 ns². There is a way to correct the bias, as described in [46].

### 3.1.3. Global fitting methods IC and VP in GF-P and GF-C modes

Figure 12 shows \( \hat{q}_1 \), \( \hat{\tau}_A \), and \( \hat{\tau}_I \) images estimated with (a1) – (a3) IC and (a4) – (a6) VP in GF-P (without HC) mode and with (b1) – (b3) IC and (b4) – (b6) VP in GF-C (with HC) mode. The estimated constants (\( \hat{\tau}_1 \), \( \hat{\tau}_2 \)) are labelled in corresponding \( \hat{q}_1 \) images. The estimations of \( q_1 \) in Region \( I_1 \) with IC in GF-P are mostly inaccurate, as shown in Fig. 12(a1) and (\( \hat{\tau}_1 \), \( \hat{\tau}_2 \)) = (0.59, 2.74) ns. As a result, \( \hat{\tau}_A \) and \( \hat{\tau}_I \) are also not correct in Region \( I_1 \), as shown in Figs. 12(a2) and (a3). In GF-C, IC performs better with a successfully estimated Region \( I_1 \), a significantly reduced
$t_{exe}$ from 724.82 s to 11.85 s, and a reduced $MSE(q_1)$ from 0.100 to 0.017, a reduced $MSE(\tau_A)$ from 0.678 to 0.102, and a reduced $MSE(\tau_I)$ from 1.098 to 0.050, as shown in Figs. 12(b1) - (b3).

HC also accelerates VP from 3.34 s to 0.31 s with a reduced $MSE(q_1)$ from 0.033 to 0.014, a reduced $MSE(\tau_A)$ from 0.122 to 0.048, and a reduced $MSE(\tau_I)$ from 0.178 to 0.093, as shown in Figs. 12(b4) – (b6).

Although VP has some invalid estimations with $q_1 < 0$ (pixels in white) when $q_1$ is small, as shown in Figs. 12(a4) and (b4), its $\hat{\tau}_A$ and $\hat{\tau}_I$ images are accurately evaluated without invalid pixels, as shown in Figs. 12(a5), (a6), (b5), and (b6). Thus, VP in GF-C is a promising choice for fast average lifetime estimations for its short execution time ($t_{exe} = 0.31$ s).

In conclusion, HC not only accelerates analysis but also enhances accuracy (MSE). HC sorts histograms with similar profiles into a cluster and takes the average of histograms for lifetime determination, equivalent to increasing the number of photon counts and reducing noise. Therefore, the decay parameters estimated with the average cluster histogram by lifetime determination algorithms have higher accuracy than those of individual histograms. Although the decay parameters of the histograms in one cluster have a deviation from those estimated with the average cluster histogram, the results indicate that the error introduced by HC is smaller than that introduced by processing original histograms with a relatively lower photon count.

### 3.2. Experimental data

Mouse raw macrophage cells were routinely cultured in DMEM (Dulbecco’s Modified Eagle Medium) supplemented with 10% FCS (Fetal Calf Serum) under 5% CO$_2$ at 37°C. Cells were seeded on glass cover slips in 24-well plates and cultured overnight for bacterial infection. Bacteria engineered to express GFP (Green Fluorescent Protein) were harvested from an early exponential phase and added to the cells with an MOI (Multiplicity of Infection) = 100. Cells were washed with PBS (Phosphate-Buffered Saline) and stained for actin with phalloidin Alexa Flour 546 (Thermo Fisher Scientific). The scanning FLIM used in this work is LSM510 (Carl Zeiss), equipped with a TCSPC module (SPC-830, Becker & Hickl GmbH). The sample was excited by a tunable femtosecond Ti: Sapphire laser (Chameleon, Coherent) at 850 nm as a two-photon excitation source. The repetition rate is 80 MHz, and the pulse width is less than 200
fs. The emitted photons were collected through a 63× water-immersion objective lens (N.A. = 1.0) and a 500 ~ 550 nm bandpass filter and transferred into a photomultiplier tube.

Figure 13 shows the intensity image and $t_{exe}$ for all algorithms without or with HC for a TCSPC dataset. The estimations of three output types are shown as follows.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Algorithm</th>
<th>$t_{exe}$ (s)</th>
<th>Mode</th>
<th>Algorithm</th>
<th>$t_{exe}$ (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PW</td>
<td>LE-LSM</td>
<td>632.32</td>
<td>CW</td>
<td>LE-LSM</td>
<td>5.87</td>
</tr>
<tr>
<td></td>
<td>LE-IEM</td>
<td>78.05</td>
<td></td>
<td>CMM</td>
<td>0.27</td>
</tr>
<tr>
<td></td>
<td>CMM</td>
<td>0.27</td>
<td></td>
<td>CMM</td>
<td>0.20</td>
</tr>
<tr>
<td>GF-P</td>
<td>IC</td>
<td>1022.79</td>
<td>GF-C</td>
<td>IC</td>
<td>6.01</td>
</tr>
<tr>
<td></td>
<td>VP</td>
<td>12.76</td>
<td></td>
<td>VP</td>
<td>0.41</td>
</tr>
</tbody>
</table>

**Fig. 13.** Intensity image and $t_{exe}$ for all algorithms without and with HC.

3.2.1. Type 1: $\hat{q}_1$, $\hat{\tau}_1$, and $\hat{\tau}_2$ images with LE-LSM in PW and CW modes

Figure 14 shows (a) $\hat{q}_1$, (b) $\hat{\tau}_1$, and (c) $\hat{\tau}_2$ images with LE-LSM in PW, (d) - (f) the results with LE-LSM in CW, and the histograms of (g) $\hat{q}_1$, (h) $\hat{\tau}_1$, and (i) $\hat{\tau}_2$ in PW (blue) and CW (red) modes. LE-LSM shows similar lifetime estimation performances in PW and CW, whereas LE-LSM in CW ($t_{exe} = 5.87$ s) is faster than LE-LSM in PW ($t_{exe} = 632.32$ s). Therefore, LE-LSM used in CW is a better choice for Type 1.

**Fig. 14.** $\hat{q}_1$, $\hat{\tau}_1$, and $\hat{\tau}_2$ images from LE-LSM (a) - (c) without and (d) – (f) with HC. Histograms of (g) $\hat{q}_1$, (h) $\hat{\tau}_1$, and (i) $\hat{\tau}_2$ in PW (blue) and CW (red) modes.

3.2.2. Type 2: $\hat{q}_1$ image and constants ($\hat{\tau}_1$, $\hat{\tau}_2$) with IC and VP in GF-P and GF-C modes

Figure 15 shows $\hat{q}_1$ images with (a) IC and (b) VP in GF-P, (c) IC and (d) VP in GF-C. Figure 15(e) shows histograms of $\hat{q}_1$ with IC (dash blue) and VP (dash red) in GF-P and IC (solid blue) and VP (solid red) in GF-C. The constants ($\hat{\tau}_1$, $\hat{\tau}_2$) of each approach are attached in $\hat{q}_1$ images.

3.2.3. Type 3: $\hat{\tau}_A$ and $\hat{\tau}_I$ images

Figure 16 shows $\hat{\tau}_A$ images (a) – (d) without and (e) – (h) with HC. Figure 16(i) shows the histograms of $\hat{\tau}_A$ with LE-LSM and LE-IEM in PW and CW. Figure 16(j) shows the histograms of $\hat{\tau}_A$ with IC and VP in GF-P and GF-C. Figure 17 shows $\hat{\tau}_I$ images (a) – (d) without and (e) –
Fig. 15. \( \hat{q}_1 \) images from IC and VP (a) – (b) without and (c) – (d) with HC. (e) Histograms of \( \hat{q}_1 \) with IC (dash blue) and VP (dash red) in GF-P and IC (solid blue) and VP (solid red) in GF-C.

(h) with HC. Figure 17(i) shows the histograms of \( \hat{\tau}_I \) with LE-LSM and CMM in PW and CW. Figure 17(j) shows the histograms of \( \hat{\tau}_I \) with IC and VP in GF-P and GF-C.

Fig. 16. \( \hat{\tau}_A \) images from the algorithms (a) – (d) without and (e) – (h) with HC. (i) Histograms of \( \hat{\tau}_A \) with LE-LSM and LE-IEM in PW and CW. (j) Histograms of \( \hat{\tau}_A \) with IC and VP in GF-P and GF-C.

Like the conclusions drawn from simulations, LE-LSM in CW is the fastest for Type 1 with \( t_{exe} = 5.87 \) s; VP in GF-C is the fastest for Type 2 with \( t_{exe} = 0.41 \) s. For average lifetime images, VP in GF-C is the fastest for both \( \tau_A \) and \( \tau_I \) with \( t_{exe} = 0.41 \) s, LE-IEM in CW is the second one for \( \tau_A \) with \( t_{exe} = 0.94 \) s; meanwhile, CMM in CW is the fastest for \( \tau_I \) with \( t_{exe} = 0.20 \) s.
4. Conclusion

We developed a histogram clustering (HC) method to accelerate FLIM analysis. HC can improve both the speed and the accuracy for FLIM analysis by sorting histograms with similar profiles in a dataset into several clusters and significantly reducing the number of histograms to be analyzed. The HC method implements clustering with two features of a histogram. Several commonly used lifetime determination algorithms’ performances for producing decay parameter images without and with HC were compared using synthetic and experimental datasets. For different output types, the fastest FLIM analysis methods are suggested: 1) LE-LSM with HC for all lifetime component images with an execution time ($t_{exe}$) of 5.87 s, 106-fold shorter than $t_{exe}$ without HC; 2) VP with HC for constant lifetimes, $q_1$, $\tau_A$, and $\tau_I$ images with $t_{exe} = 0.41$ s, 32-fold shorter than $t_{exe}$ without HC; 3) LE-IEM with HC as the second choice for $\tau_A$ with $t_{exe} = 0.94$ s, 78-fold shorter than $t_{exe}$ without HC, and CMM as the second choice for $\tau_I$ with $t_{exe} = 0.2$ s without or with HC (biased if the largest lifetime > $T/4$). The analysis was conducted in Matlab, and it can be translated to C or other environments to speed up the analysis. We believe the proposed HC method can benefit applications demanding real-time FLIM such as clinical diagnosis and fast screening.

Funding. Science Foundation of the Chinese Academy of Sciences (CXJJ-21S006); Strategic Priority Research Program of Chinese Academy of Sciences (XDA25030900); National Natural Science Foundation of China (12075312); Medical Research Scotland (1179-2017); Engineering and Physical Sciences Research Council (EP/L01596X/1).

Acknowledgments. N. Sapermsap acknowledges the Development and Promotion of Science and Technology Talents (DPST) Project under the Institute for the Promotion of Teaching Science and Technology (IPST), Thailand, for a PhD scholarship.

Disclosures. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are not publicly available and can be made available by the authors without undue reservation.

Supplemental document. See Supplement 1 for supporting content.

References