

A decomposition algorithm for streak camera data

Kaan Oguzhan,^{a,1} Lucas Ranc,^a Livio Verra,^{b,2} Allen Caldwell^a

^a*Max Planck Institute for Physics, Munich, Germany*

^b*CERN, Geneva, Switzerland*

E-mail: kaan.oguzhan@tum.de, ranc@mpp.mpg.de, caldwell@mpp.mpg.de,
livio.verra@cern.ch

ABSTRACT: We describe a novel reconstruction algorithm for time-resolved images obtained using a streak camera. This algorithm operates by decomposing a recorded image into a set of individual photoelectron-induced signals, thereby providing a powerful method for streak camera image reconstruction. This deconstruction allows for a standard statistical analysis of the resulting image. We demonstrate the effectiveness of this technique by analyzing the temporal spacing between the emitted fs-long laser pulse and its succeeding first, second, and third reflections within a thick glass captured by a streak image.

KEYWORDS: Beam-line instrumentation, Data processing methods, Analysis and statistical methods

¹Corresponding author.

²Now at: INFN Laboratori Nazionali di Frascati, Frascati, Italy

Contents

1	Introduction	1
2	Experimental Setup	2
2.1	Operational Principles of Hamamatsu Streak Camera C10910	2
2.2	Laser System	3
2.3	Light Reflection Setup	4
2.4	Additional Dataset	5
3	Photoelectron Decomposition Method	5
3.1	Isolating Individual Photoelectrons Using Binary Filtering	6
3.2	Analysis of Individual Photoelectron Clusters	9
3.3	Photoelectron decomposition algorithm (PDA)	11
3.4	Reversible decomposition	12
4	Timing measurements	13
4.1	Finding the location of features in time	14
4.2	Time resolution	14
5	Conclusions	17

1 Introduction

Streak cameras are widely used instruments in the particle accelerator community for the acquisition of time-resolved signals [1–8]. The operational principle involves incident light signals, commonly photons, impinging on a cathode, thereby generating photoelectrons. These photoelectrons are then accelerated and then deflected vertically by an angle which depends on their arrival time. Deflected photoelectrons emit a light signal upon reaching a phosphor screen. This signal, captured by a CMOS camera, enables timing measurements of incident signals based on their vertical image position. The typical image reconstruction process is based on the light signal’s amplitude as measured by the CMOS camera. In this article, we describe a novel algorithm for reconstructing the photoelectron distribution based on the recorded camera image. This is accomplished by initially identifying the characteristics of the signals generated by individual photoelectrons, and then decomposing the overall image in term of photoelectrons. This makes an analysis possible based on standard statistical techniques, so that uncertainty estimation on a picture-by-picture basis is available.

We have performed our studies in the context of the AWAKE experimental setup [9–12]. The AWAKE Collaboration pursues the demonstration of electron acceleration in a plasma wakefield driven by protons. In the context of AWAKE, streak cameras are used for capturing time-resolved images of the proton beam as well as the produced plasma wakefield structure. The resulting

time-resolved images are used for studying a variety of features of the modulation of the proton bunch in the plasma [1, 12–15]. An important component of the setup is a high-power laser [16]. This laser produces a high-intensity pulse that ionizes a 10-m-long column of Rb vapor, generating the plasma, and a timing reference signal that is sent to the streak camera. An additional specially prepared path for the laser is utilized for the studies described in this article.

In the following, we describe the experimental setup, with a dedicated focus on the streak camera. The specific setup of the laser system that was employed for the analyzed data is described in [12] and briefly reviewed. We then describe the method developed for our study and show an example of image deconvolution. The timing reconstruction based on the deconvolved image is then described, and results are shown on the time reconstruction and uncertainty estimate.

2 Experimental Setup

2.1 Operational Principles of Hamamatsu Streak Camera C10910

The Hamamatsu Streak Camera C10910 is the imaging instrument used to record images for the datasets of this paper [17]. The streak camera is designed to detect photons ranging from the UV to the near-infrared spectrum and records them as 2D images with one spatial dimension and the temporal dimension as axes. We denote the temporal dimension as the vertical dimension on the resulting images.

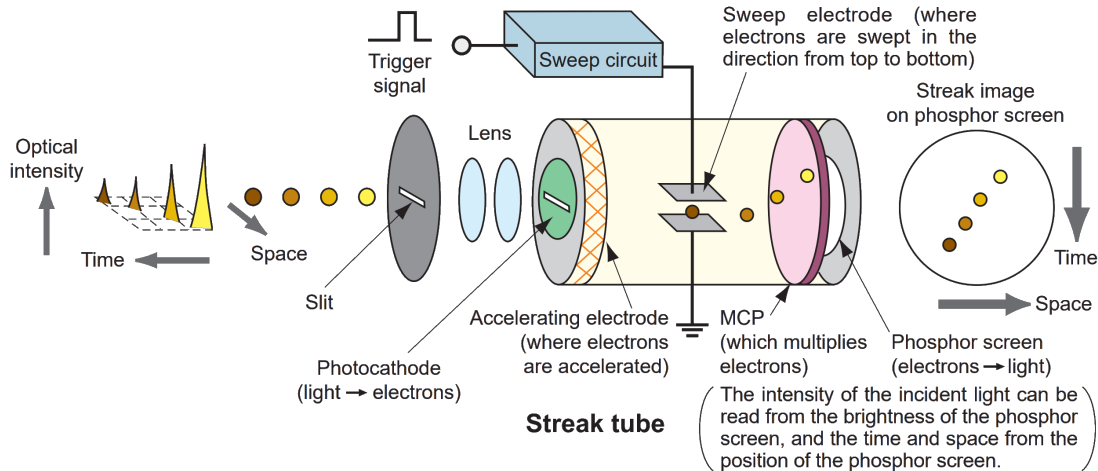


Figure 1: Illustration of the working principle of the Hamamatsu Streak Camera C10910. The diagram shows four light pulses with varying amplitudes and separated in time passing through the components of the streak camera before reaching the phosphor screen. This figure is taken from the Hamamatsu Manual [17].

The operating principle of the streak camera, as illustrated in Fig. 1, involves several stages. Initially, the light intersects with a narrow slit whose width is set to $20 \mu\text{m}$. Under the guidance of the imaging system, this light is directed to the photocathode, which converts a fraction of the incoming photons into photoelectrons. These photoelectrons are then accelerated towards the end

of the streak tube and deflected in the direction perpendicular to the slit. After amplification, they impinge on the phosphor screen, generating light pulses. The deflection process's speed is adjustable and was set such that the time span of an image corresponds to either 73 ps or 210 ps for the analysed data.

The phosphorescent images are captured by a CMOS readout camera, which is located behind the phosphor screen. Due to the multiplication of the photoelectrons by the multi-channel plate (MCP) and the light production and emission process of the phosphor screen, individual photoelectrons are observed as clusters spanning multiple pixels in the CMOS readout camera. The camera has 512 pixels in the vertical (time) direction and 672 pixels in the horizontal direction. One pixel in the vertical direction therefore corresponds to 0.143(0.410) ps for the 73(210) ps image time windows.

The streak camera is triggered by an external signal, and this is used to initiate the voltage sweep across the sweep electrode. In previous studies, the time jitter of the triggering mechanism has been determined to be 4.8 ps (rms) [12], implying that the resulting images jitter in the vertical dimension (time dimension) on a picture-by-picture basis by at least this amount. The intrinsic time resolution of the streak camera is much better than 4.8 ps so that relative time measurements amongst components of a single image can be determined with higher precision. Our algorithm aims to extract relative time measurements with the highest achievable precision using the streak camera.

2.2 Laser System

The data used for the development of our algorithm was generated by a TW-class Ti:Sapphire laser system [12]. The laser system amplifies an erbium-doped fiber oscillator with frequency-doubled 780 nm output to produce pulses with an energy up to 650 mJ at 10Hz and a duration of 110 fs. For the operation of these studies, an attenuated compressed laser pulse was used. The laser pulse duration shown Fig. 2 was obtained by a second-harmonic auto-correlator (Bonsai from Amplitude Company). More information on the laser system can be found in [18].

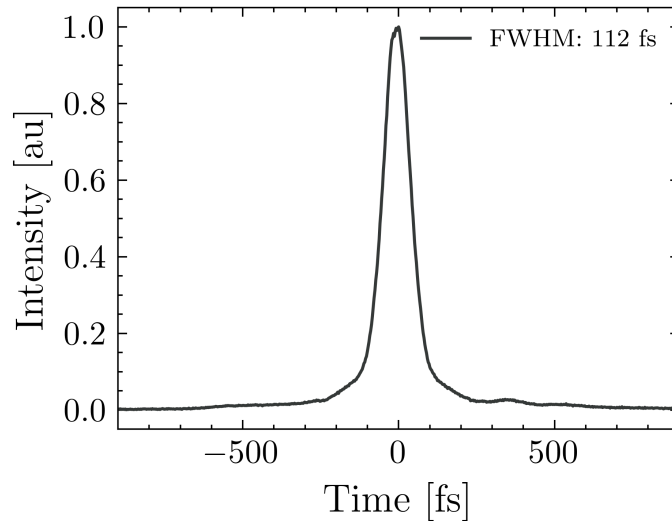


Figure 2: Temporal profile of the laser pulse obtained by a second-harmonic correlator

2.3 Light Reflection Setup

The primary dataset used in the development of our streak camera signal reconstruction algorithm was acquired in the test setup described in [12]. The experimental setup was a subsystem of the AWAKE experiment.

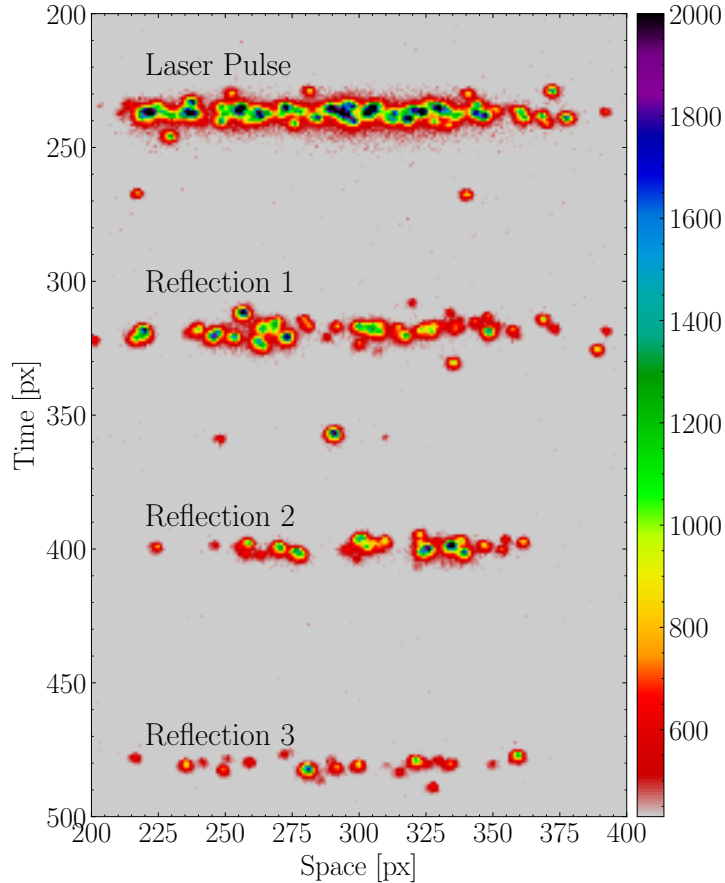


Figure 3: Streak camera image of the primary laser pulse and the first three reflections inside the glass plate.

The laser light path to the streak camera included a ~ 5 mm-thick piece of glass to obtain multiple surface reflections with fixed time intervals. In total, 1070 laser reflection images were acquired with this setup. The sweep time used for this dataset¹ was 210 ps, so that images resulting from internal reflections in the glass are expected to be separated by ~ 40 ps, equivalent to ~ 100 pixels (px). Fig. 3 shows a streak camera image from this data set. We select a 200-px-wide region of interest in the spatial direction to focus on the laser pulse and its reflections. Within the ROI, we observe the primary laser pulse at approximately pixel coordinate 237 on the time axis, along with consecutive reflection pulses generated from up to three round-trips through the glass at around time pixel coordinates of 318, 400, and 480, respectively. We convert from pixel coordinate to time in picoseconds as described below.

¹We note that the sweep time reported in [12] is not correct.

The expected delay between the reflections could not be determined precisely based on measurements of the glass thickness and its refraction index. However, by comparing the measured time differences between different reflections, we are able to determine the uncertainty of the time measurements achievable by the streak camera, without knowing the exact time spacing between the reflections.

When analysing the obtained streak camera images, we observed that the isolated signals in the streak camera images typically appeared as either single pixels or as small pixel clusters with pixel amplitudes near the CMOS baseline. The single pixels were interpreted as noise fluctuations, whereas the small clusters were identified as clusters generated by single photoelectrons. The features of the clusters are presumably affected by numerous factors such as the MCP pore size, the amplification settings of the MCP, the image spread created in the phosphor screen, the CMOS sensors pixel size, etc. These factors affected the achievable time resolution of the streak camera. In order to apply the algorithm described below to data sets with different streak camera operational settings, the photoelectron characterization steps will need to be repeated.

In our study of images taken with different sweep speeds, we observed that the streak camera time-window size has little effect on the pixel size of clusters produced by single photoelectrons. As a consequence, the algorithm operates at the pixel level and as the cluster size in pixels remains unchanged across different sweep speeds.

2.4 Additional Dataset

We have also made use of an additional dataset with 73 ps sweep speed. This allowed us to compare images acquired with different sweep speeds, and provided separate calibration data for our algorithms, as explained below. The laser pulse, used as a timing reference in this dataset, did not include the glass piece and contained no reflections of the laser pulse. An artificial 0-distance reflection was simulated in the data by dividing the recorded laser pulse image into two sets of pixels along the spatial direction. In this case, we expect the same time to be measured in each half and a comparison of the recorded times allows for a study of the uncertainties in the time reconstruction. A total of 200 images were recorded in this set of conditions.

3 Photoelectron Decomposition Method

An overview of our Photoelectron Decomposition Algorithm (PDA) is given in [Fig. 4](#). In the following subsections, we describe the different steps in the algorithm. The performance is then described in the following section.

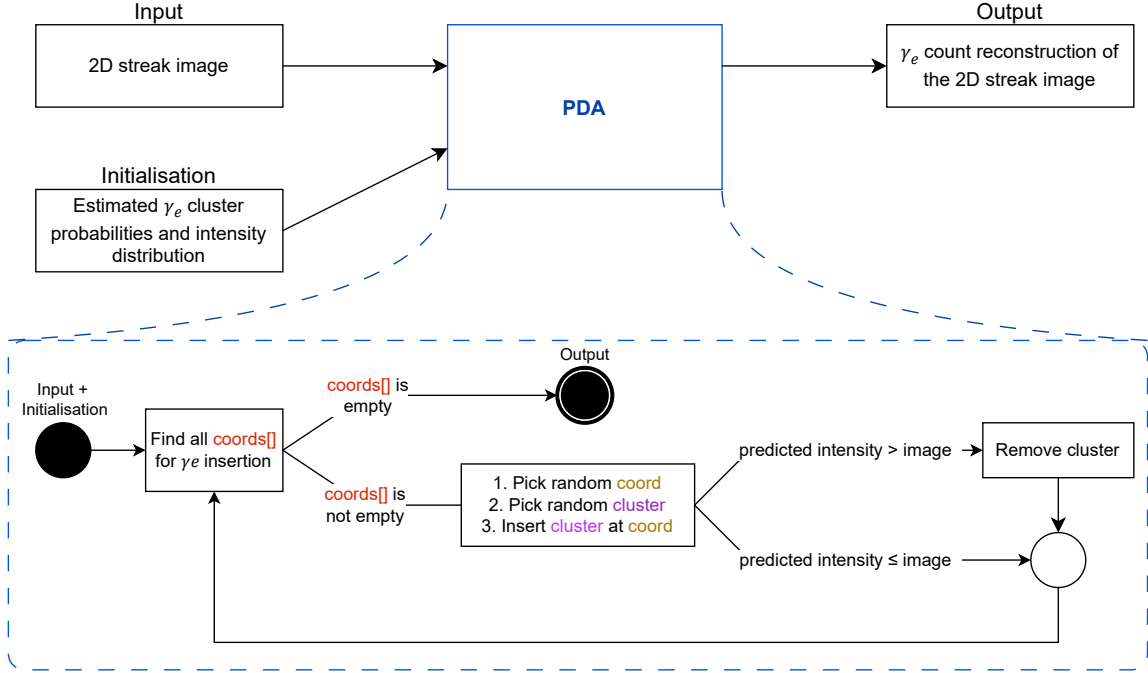


Figure 4: Overview of the Photoelectron Decomposition Algorithm. The algorithm takes as input the recorded image and the photoelectron cluster probabilities. The algorithm then performs a reconstruction of the photoelectron distribution for the recorded image. The symbol γ_e stands for photoelectron.

3.1 Isolating Individual Photoelectrons Using Binary Filtering

In a first step, we employ a binary filter to isolate as many single-photoelectron induced clusters as possible from the streak camera image. To achieve this, we apply a cutoff determined from the pedestal value of the streak camera image. The individual pixel amplitude distribution for one image in our dataset is shown in Fig. 5. A blow-up of the distribution around the pedestal value taken from a streak camera image region outside the ROI is also shown. We chose a threshold amplitude value of 430 counts for the binary filtering. This allows a reduction of the bulk of the pedestal fluctuations while preserving the bulk of photoelectron generated signals.

The operation of the binary filter [19] is rather straightforward. It assigns pixel values of 1(0), depending on whether their amplitude is above(below) a predefined threshold. We demonstrate this filtering on the streak camera image shown in Fig. 3 as a filtered image in Fig. 6. All pixels above the threshold are displayed against a white background. The two shadings (black and grey) are used to distinguish clusters associated with single photoelectrons as explained below.

The next step is to apply a connected component labeling algorithm [20] to the pixels with value 1. This algorithm identifies and groups all pixels with neighboring non-zero pixels along the temporal and spatial axes into clusters and assigns a unique label to each cluster, effectively forming distinct, traceable entities. The resulting cluster size distribution along the spatial and temporal axes are shown in Fig. 7. Two clear and identifiable cluster size regions are visible. The peak at a cluster size of one pixel in the temporal and spatial directions is from noisy pixels with amplitude above

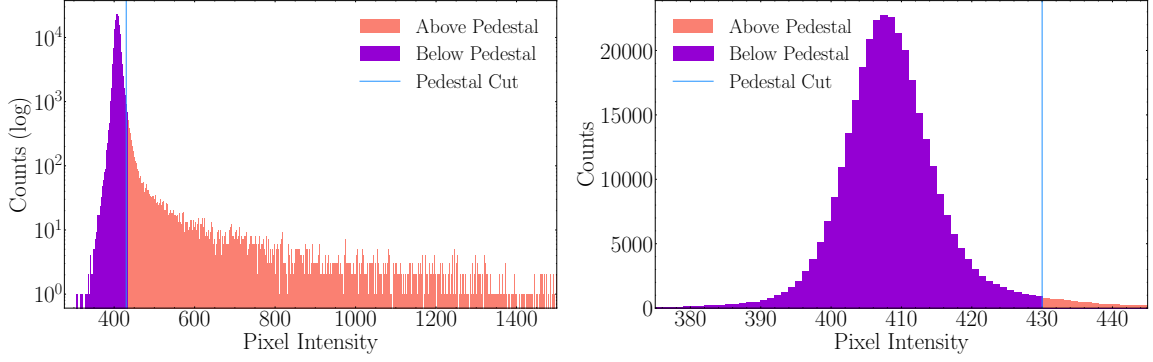


Figure 5: Histogram of the pixel amplitudes for one image in our dataset. The left panel shows the full distribution in log scale, while the right panel shows a blow-up of the pedestal region using linear scale from a regions outside the ROI. The vertical line indicates the threshold value of the pedestal used for binary filtering.

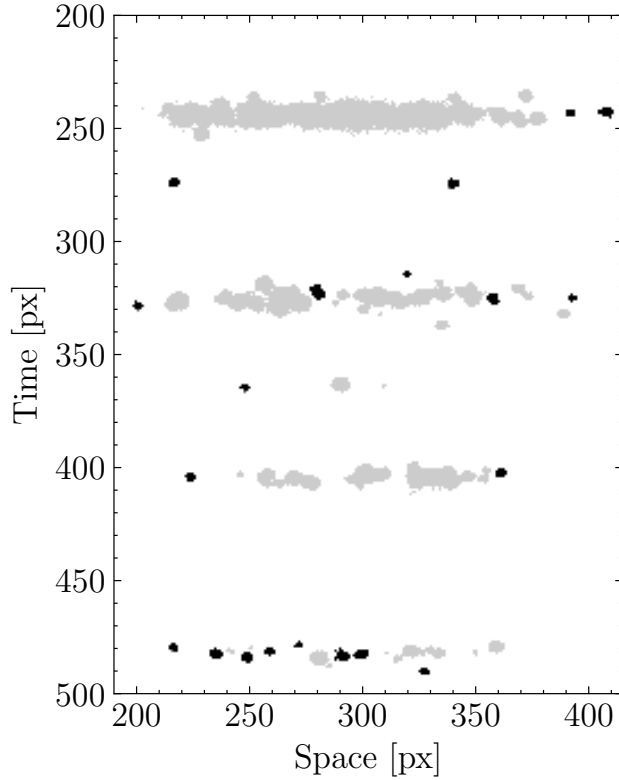


Figure 6: Pixels in the image shown in Fig. 3 passing the binary thresholding test are shown. The pixels belonging to clusters identified as resulting from single photoelectron are shown in the darker shading, while the lighter shading represents pixels from clusters that are too big or too small to be identified as resulting from single photoelectrons.

threshold. We identify the larger island with ~ 5 pixels along the spatial and temporal axes as arising from single photoelectrons.

To select clusters identified as arising from single photoelectrons, we employ multiple limitations

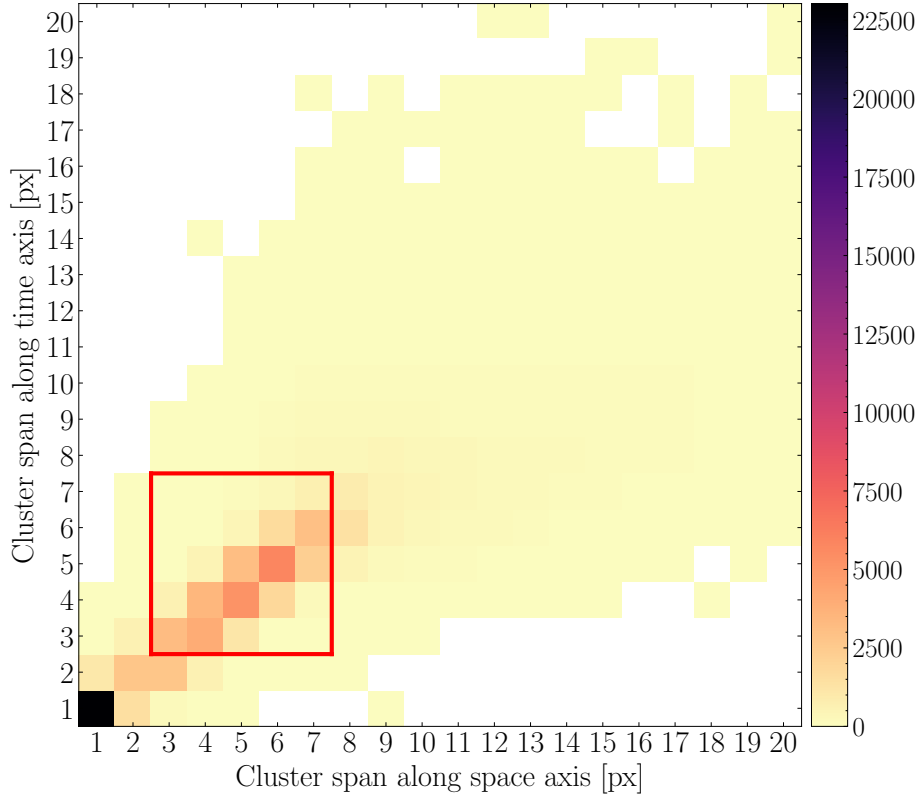


Figure 7: Heatmap of the cluster sizes along space and time axes of the streak image. The color scale gives the number of clusters found with a particular size. The red box indicates the cluster size region used to identify single photoelectrons.

on the cluster size. Firstly, we restricted the cluster size to be between 3 to 7 pixels in both the spatial and temporal axes as shown with the red square in Fig. 7. Secondly, we discarded any cluster that has less than 8 total pixels above the pedestal. This requirement was needed to eliminate a special class of noise clusters appearing as a diagonal feature in the image. Lastly, we discarded the cluster if any pixel had an amplitude more than 500 counts above the pedestal threshold, again to eliminate clusters deemed to be caused by noise.

Fig. 6 shows the result of this filtering on a sample image, where the black pixels are the pixels that passed the filtering and the grey pixels are the pixels that failed the filtering. From the 1070 images in our primary dataset, we detected 23888 single photoelectron clusters. This significant dataset allowed for a detailed study clusters identified as resulting from single photoelectrons.

3.2 Analysis of Individual Photoelectron Clusters

To analyze the amplitude distribution caused by individual photoelectrons, all clusters from the laser reflection images were examined. The two-dimensional clusters were transformed into one-dimensional clusters by summing pixels along the space axis. We then categorized the clusters based on their width along the time axis. As a result of this process we obtained five separate classes each with different pixel widths. Additionally, for each category we assigned class probabilities by dividing the number of observed clusters in the class by the total number of observed clusters. [Table 1](#) shows associated counts and probabilities for each category. 95.3 % of clusters identified as resulting from single photoelectrons have widths along the time axis of 3 – 5 pixels.

<i>Cluster Class</i>	3 px	4 px	5 px	6 px	7 px
Detected Count	6099	11497	6283	881	309
Probability	0.243	0.459	0.251	0.035	0.012

Table 1: Photoelectron cluster size distribution along the time axis. ‘Detected Count’ gives the number of clusters detected for each cluster size over 1070 images and ‘Probability’ gives the associated probabilities for each cluster size.

For each of these classes, we calculated an average amplitude profile along the time axis and its fluctuations. These profiles are shown in [Fig. 8](#) for each class. The profiles were centered within 9-pixel windows to allow for a visual comparison between the different classes. The error bar is the uncertainty on the average calculated from the RMS divided by the square root of the number of clusters in that class. We next use these cluster profiles and their occurrence probabilities to decompose streak camera images into distributions of photoelectrons.

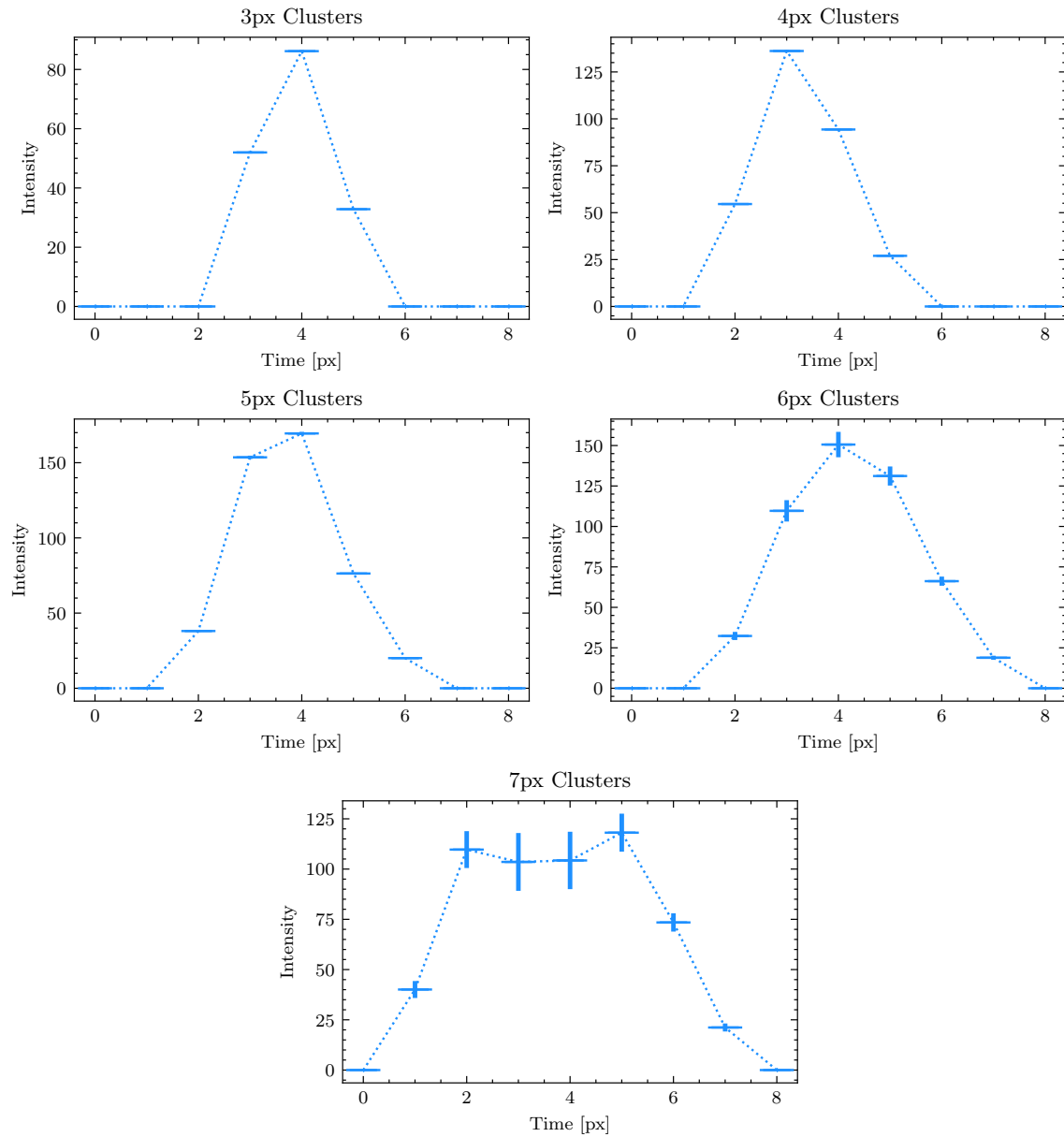


Figure 8: Averaged time profiles of clusters identified as resulting from single photoelectrons, separated into clusters with different numbers of pixels as indicated in the legends. The amplitude values are averaged and centered within 9-pixel windows. The error bars represent the RMS of the amplitudes recorded at the different cluster pixel locations divided by the square root of the number of clusters in that class.

3.3 Photoelectron decomposition algorithm (PDA)

Input : Amplitude distribution of clusters along the time axis

Output : Individual photoelectron counts

Function `reconstructPhotoelectrons` (*amplitude_distribution*) :

```
cluster_probabilities[] ← Probabilities for photoelectron clusters of different widths;
predicted_amplitude[] ← Zeros with size equal to amplitude_distribution;
photoelectron_distribution[] ← Zeros with size equal to amplitude_distribution;
insertion_locations[] ← find_photoelectron_insertion_locations(predicted_amplitude[]);

while  $sum(insertion\_locations) \neq 0$  do
    location ← pick_random_location(insertion_locations[]);
    cluster[] ← pick_random_cluster(cluster_probabilities[]);
    next_amplitude_prediction[] ← insert_cluster(predicted_amplitude[], location,
        cluster[]);
    if  $all(next\_amplitude\_prediction[] \leq amplitude\_distribution[])$  then
        photoelectron_distribution[] ← increment_photoelectron(location);
        predicted_amplitude[] ← next_amplitude_prediction[];
    end
    insertion_locations[] ←
        find_photoelectron_insertion_locations(predicted_amplitude[]);
end
return photoelectron_distribution;
```

Algorithm 1: Photoelectron Decomposition Algorithm

The photoelectron decomposition algorithm is applied to clusters that are larger in size than single photoelectrons. These clusters are first summed along the spatial direction resulting in a one-dimensional amplitude distribution along the temporal axis. Our algorithm then builds up a simulated amplitude distribution based on summing amplitudes distributions from single photoelectrons placed at different locations along the temporal direction. The algorithm is designed to have the simulated distribution reproduce the observed distribution. The end result of the process is a distribution of photoelectrons along the temporal direction.

The algorithm begins with an empty simulated amplitude distribution. A pixel coordinate for insertion of a photoelectron is chosen at random, and the cluster size is then randomly chosen according to the probabilities given in [Table 1](#). An attempt is then made to add the average amplitude profile to the simulated distribution. In this process, the photoelectron amplitude profile is centered as follows: if the cluster has an odd number of pixels, the amplitude profile is centered on the chosen pixel location, while, if the cluster has an even number of pixels, the profile is randomly centered on one of the two central pixels with equal probability for each. If the insertion results in amplitudes greater than the observed amplitude for any pixel, the insertion is rejected. Otherwise, the insertion

is accepted and the algorithm moves to the next randomly picked location. The algorithm terminates when it cannot detect any possible location for photoelectron insertion. A pseudo-code of the algorithm is given in [algorithm 1](#).

3.4 Reversible decomposition

It is possible to convert the pixel amplitudes to photoelectron counts and vice versa. This provides a valuable check for our algorithm. As an example, we show a detailed view in the 30-pixel-wide (along the time axis) window containing a laser pulse image in [Fig. 9](#) and proceed to analyze this image. As described above, our algorithm works in a stochastic manner, and successive decompositions will produce different photoelectron distributions.

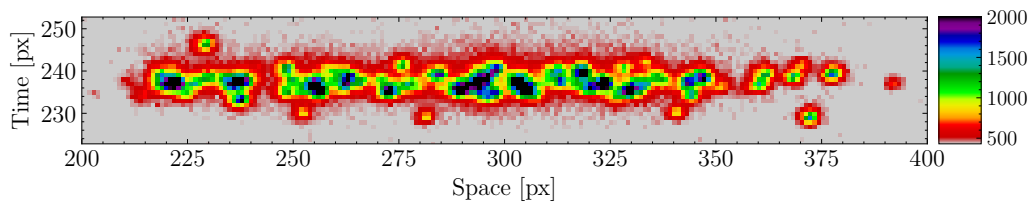


Figure 9: Expanded view of the signal region produced by a laser pulse. The signal amplitude is given in color scale.

The amplitude distribution for this image, integrated over the spatial direction, is shown in the upper panel in [Fig. 10](#). The result of the photoelectron decomposition algorithm is also shown. As seen in the Figure, the photoelectron distribution is somewhat narrower than the observed amplitude distribution, as the photoelectron clusters broaden the image. The decomposition process effectively deconvolves the amplitude distribution and produces a somewhat narrower distribution.

We can perform the procedure in reverse by using the photoelectron cluster distribution, selecting from our cluster spectra according to their probabilities, and creating an amplitude distribution. We have repeated the process of recreating the amplitude distribution 10000 times for the image shown in the upper panel of [Fig. 10](#), based on one photoelectron decomposition. The results are shown in the lower panel of the figure, where the observed amplitude distribution is given by the diamond shaped symbols and the average reconstructed image is shown in horizontal bars. The RMS of the amplitude distributions from the 10000 repetitions is shown as the error bar on the prediction and is typically within the size of the symbol. As can be seen, the average agrees well with the observed distribution, and the spread in the predicted signal shape is quite small.

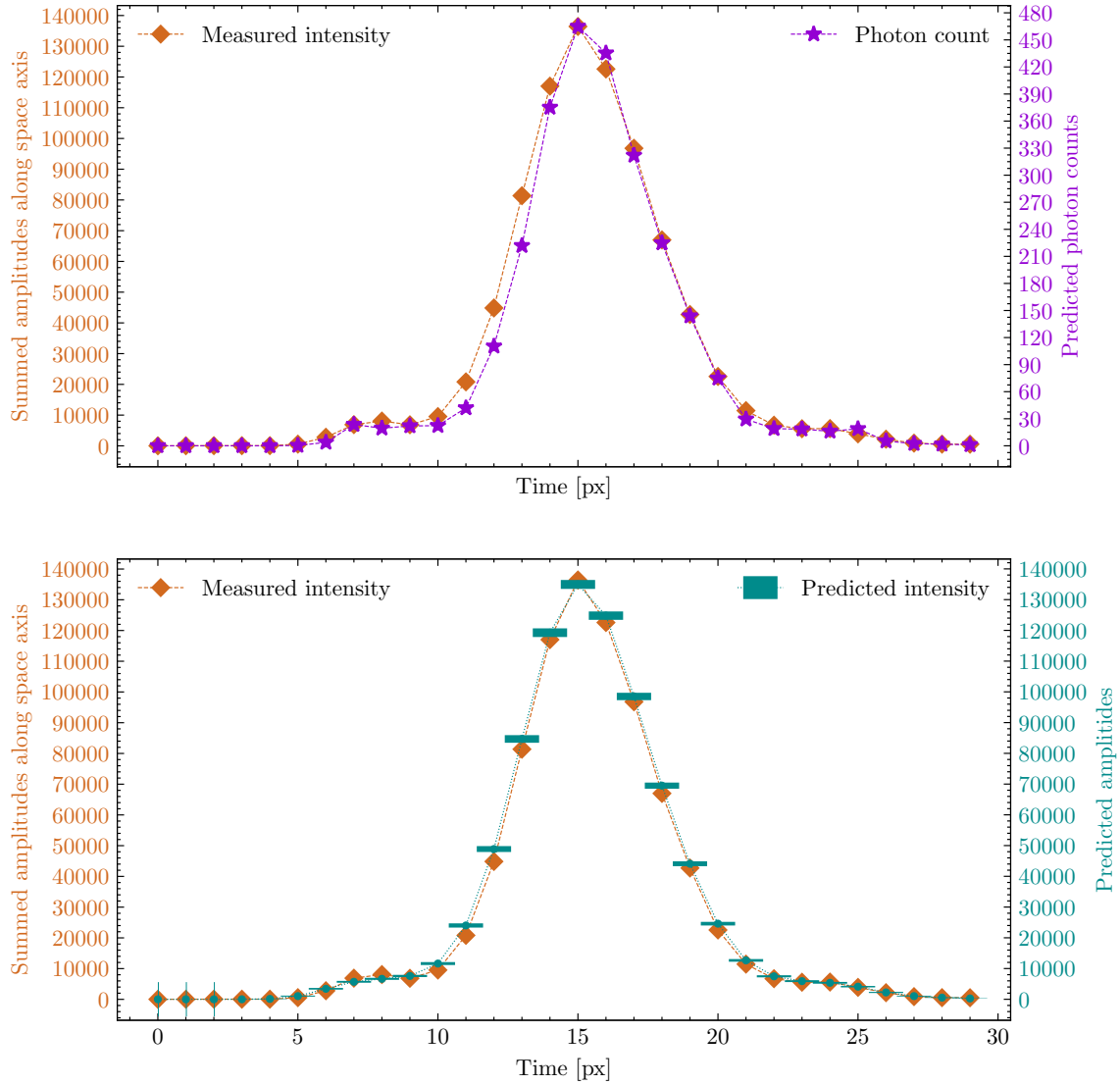


Figure 10: Upper panel: Measured time profile and extracted number of photoelectrons for the image shown in Fig. 9. The horizontal axis shows time in pixel units, the left hand side vertical axis gives the measured amplitudes and the right hand side vertical axis gives the extracted photoelectron counts. Lower panel: Comparison between predicted amplitudes to the measured amplitudes using our reversed photoelectron decomposition algorithm. The error bars show the RMS of the predicted amplitudes.

4 Timing measurements

The primary use of the streak camera is to extract timing information. We have studied the potential timing resolution of features within individual streak images. This has been accomplished through the usage of the photoelectron decomposition method on two distinct datasets: the ‘reflections’ dataset, which used a 210 ps sweep time, and a dataset with 73 ps sweep time.

4.1 Finding the location of features in time

Our algorithm provides the pixel location of photoelectrons along the time axis of the image and thereby enables the application of standard statistical methods to estimate uncertainties for timing feature extraction. Typical approaches for finding feature location, e.g., fitting a Gaussian function to the amplitude profile of the feature, does not allow the calculation of the uncertainty on the time measurement on a per image basis or a per feature basis.

We calculate a feature time using a weighted average of the photoelectron pixel locations, where the weights are the photoelectron counts. The weighted average is given by

$$\bar{t}_{px} = \frac{\sum_{i=1}^n w_i t_i}{\sum_{i=1}^n w_i}, \quad (4.1)$$

where t_i is the pixel number of the i -th pixel along the time axis in the image and w_i is corresponding photoelectron count. The averaged pixel value is then converted to time using the conversion based on the sweep time window. The uncertainty on this time estimate is expected to scale inversely with the square root of the total number of photoelectrons.

We use this scheme to analyze the data from our two datasets. As a first step, we need to locate a ROI within the image to be analyzed. We do this by fitting Gaussian functions to temporal amplitude profiles summed among the space axis and choosing a suitably broad region around the peak pixel. The ROI shown in Fig. 9 was chosen by adding ± 15 px to the center pixel from the Gaussian fit. We have repeated this process for the ROIs found in each image of our datasets and applied our time reconstruction algorithm to these ROIs.

In the upper panel of Fig. 11 we show the reconstructed times as a function of image number for all 1070 images in our glass pane reflection dataset. The top row of times are for the laser pulse, while the three rows below are from (1-3) sets of consecutive reflections in the glass. The variation of the recorded times of each pulse are due the trigger jitter of the streak camera. This is shown by the fact that the fluctuations of each pulse are correlated with all of its reflections.

We use the notation Δt_1 to represent the time difference between the laser pulse and Reflection 1. Similarly, $\Delta t_2, \Delta t_3$ give the time differences between the reflections with 1 and 2 round trips, and the reflections with 2 and 3 round trips in the glass, respectively. The mean values are

$$\begin{aligned} \overline{\Delta t_1} &= 80.72px(33.11ps) \quad RMS = 0.53px(0.22ps) \\ \overline{\Delta t_2} &= 80.69px(33.10ps) \quad RMS = 0.77px(0.31ps) \\ \overline{\Delta t_3} &= 80.67px(33.09ps) \quad RMS = 1.31px(0.54ps), \end{aligned}$$

in excellent agreement with each other.

We observed that the time differences between the laser pulse and its reflections are measured with much higher precision than the absolute times of the features with respect to the streak camera trigger, which is in agreement with the ~ 4.8 ps (rms) jitter found in [12].

4.2 Time resolution

We used the following simple parametrization for evaluating the uncertainty on the time measured for a given image feature:

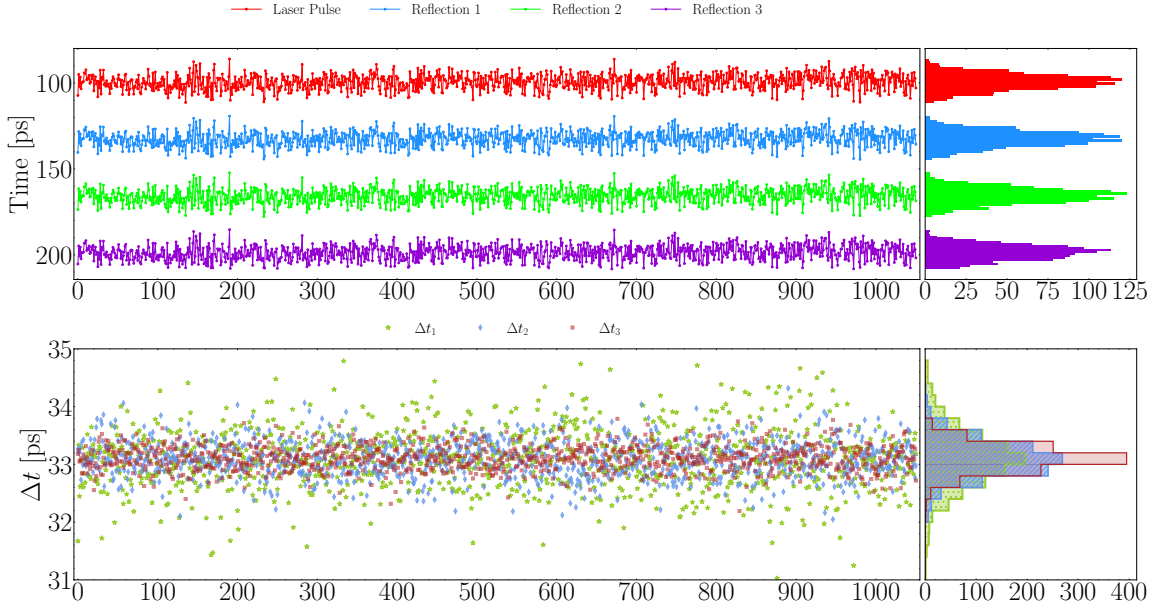


Figure 11: Top panel: Reconstructed timings for the laser pulse and its three reflections (1-3) from multiple round trips through the glass slab, displaying all images from the reflection dataset. Bottom panel: Time differences between the measured arrival times of the laser pulse and its subsequent three reflections.

$$\sigma_t = \frac{\sigma_{\text{eff}}}{\sqrt{n_\gamma}} \quad (4.2)$$

where σ_{eff} is an effective resolution per photoelectron and $n_\gamma = \sum_{i=1}^n w_i$ is the number of photoelectrons found in the feature for which we are extracting a time measurement. The parametrization presented in this work is applicable when n_γ is neither too small nor too large. We have tested its effectiveness for $400 \lesssim n_\gamma \lesssim 3000$. In our experimental setup, the average number of photoelectrons reconstructed for the laser pulse was approximately 2800, which decreased by a factor of approximately 0.55 for each round-trip through the glass. The value for σ_{eff} was determined from the primary laser pulse by splitting the image into two parts along the space axis and analyzing the resulting reconstructed time distribution. This procedure yielded $\sigma_{\text{eff}} = 18.28$ px. Although this is an effective per photoelectron resolution, it only applies when the number of photoelectrons is large enough as the smallest number of n_γ used in our tests was approximately 400.

As a test of our parametrization for the resolution, we calculated the residual distributions for the different Δt . The residual for Δt_1 for image i is defined as

$$r_{1,i} = \frac{\Delta t_{1,i} - \overline{\Delta t_1}}{\sqrt{\sigma_{t,i,\text{LaserPulse}}^2 + \sigma_{t,i,\text{Reflection1}}^2}} \quad (4.3)$$

where the index i gives the image number. $\overline{\Delta t_1}$ is the mean of the $\Delta t_{1,i}$ and was reported above. The corresponding residuals were calculated for Δt_2 and Δt_3 . Fig. 12 shows the residual distributions of all three Δt .

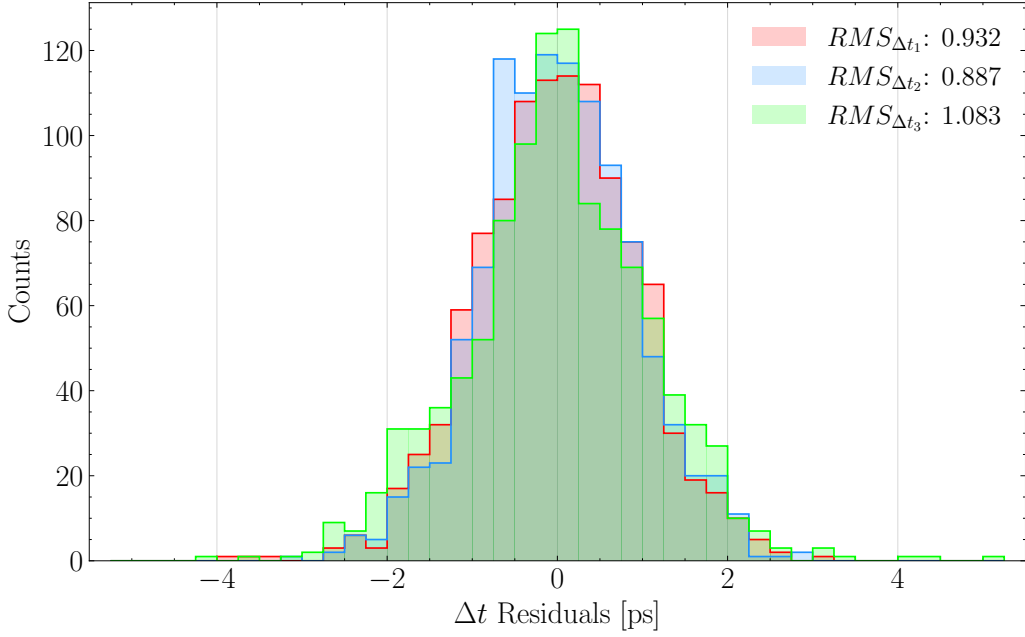


Figure 12: Residual distribution using the uncertainty parametrization Eq. 4.2.

As can be seen in the figure, the residual distribution is centered on 0 by construction, and has a width close to 1, indicating that the uncertainty on the time measurement is well estimated. Given that these results were extracted using a dataset with a 210 ps sweep time window, we can report the timing resolution for this sweep window as

$$\sigma_t = \frac{7.50}{\sqrt{n_\gamma}} \text{ ps} .$$

for $400 \leq n_\gamma \leq 3000$. I.e., a timing resolution of 140 fs is reached for $n_\gamma = 3000$ and this sweep window.

As discussed above, the resolution of the streak camera is due primarily to the single photoelectron cluster size in pixels, so that we expect a scaling of the resolution approximately with the length of the sweep time window. Using a 73 ps time window, our parametrization then gives a time resolution of

$$\sigma_t = \frac{2.61}{\sqrt{n_\gamma}} \text{ ps} .$$

We have tested this prediction by analyzing an AWAKE proton bunch modulation dataset with the streak camera window sweep time set to 73 ps. The full photoelectron cluster characterization was carried out for the streak camera settings used in this dataset. The photoelectron decomposition was then applied to the reference laser pulse in the acquired images. We used the technique of splitting the laser pulse image spatially in two parts and measuring the time difference found between the two halves of the pulse. The dataset consisted of 200 images and the typical number of photoelectrons reconstructed from the laser pulse was 800, with each half having approximately 400 photoelectrons. This yielded a predicted resolution for each slice of the laser pulse of 130 fs.

Given that the uncertainties on the slices are expected to be uncorrelated, the uncertainty on the time difference between the two slices is expected to be $\sqrt{2}$ times larger than the uncertainty on each slice, which is approximately 180 fs. We measured the RMS of the time difference between the slices of the laser pulse as 200 fs, which agrees well with the predicted value, so that we are confident our parametrization of the time resolution in terms of pixel quantities can be transformed to a time resolution using the sweep time window.

5 Conclusions

We have introduced a novel algorithm that decomposes streak camera images into distributions of photoelectrons. This algorithm relies on an image deconvolution technique that uses premeasured photoelectron profiles to create deconvolution kernels. The image clusters formed by single photoelectrons are characterized in terms of widths in pixels and amplitude. The cluster widths are found to be largely independent of the sweep speed of the streak camera. Performing the image decomposition into a distribution of photoelectrons in space and time enables the application of traditional statistical analyses, including counting statistics, on the resultant image features and to report individual uncertainty measurement for each feature in the image.

We tested our algorithm and its uncertainty prediction for timing measurements using a dataset where laser pulses, separated by a fixed time interval, were available as well as with AWAKE data sets. The dataset with multiple laser pulse features was produced by introducing a glass pane in the path of a laser pulse, generating several image features due to reflections in the glass. The time differences between the different reflections were then analyzed. For the AWAKE data set, the image from a single laser pulse was spatially split and the timing measurements of the two parts compared. The time resolution was found to be $\sigma_t = 7.50/\sqrt{n_\gamma}$ ps, for 210 ps sweep time and $\sigma_t = 2.61/\sqrt{n_\gamma}$ ps, for 73 ps sweep time. This simple formula was verified for $400 \leq n_\gamma \leq 3000$.

Acknowledgments

We would like to express our gratitude to the AWAKE Collaboration for providing the glass pane reflection dataset and the proton beam dataset. We expressly thank Fabian Batsch for his help in understanding the specifics of the reflection dataset which he set up and generated.

References

- [1] L. Verra et al. (AWAKE Collaboration), *Controlled Growth of the Self-Modulation of a Relativistic Proton Bunch in Plasma*, *Phys. Rev. Lett.* **129** (2022) 024802 [2203.13752].
- [2] A.-M. Bachmann and P. Muggli, *Beam Diagnostics in the Advanced Plasma Wakefield Experiment AWAKE*, *IBIC* (2020) MOAO01.
- [3] M.J. Hogan, C.E. Clayton, C. Huang, P. Muggli, S. Wang, B.E. Blue et al., *Ultrarelativistic-positron-beam transport through meter-scale plasmas*, *Phys. Rev. Lett.* **90** (2003) 205002.
- [4] K. Rieger, A. Caldwell, O. Reimann and P. Muggli, *GHz modulation detection using a streak camera: Suitability of streak cameras in the AWAKE experiment*, *Rev. Sci. Instrum.* **88** (2017) 025110.

- [5] A.-M. Bachmann and P. Muggli, *Determination of the Charge per Micro-Bunch of a Self-Modulated Proton Bunch using a Streak Camera*, *J. Phys. Conf. Ser.* **1596** (2020) 012005.
- [6] E. Rossa, C. Bovet, L. Disdier, F. Madeline and J.-J. Savioz, *Real Time Measurement of Bunch Instabilities in LEP in three Dimensions using a Streak Camera*, Conference Proceedings, Editions Frontieres, Paris, France (1992).
- [7] S. Wang and D. Rubin, *Measurement of the beam yz crabbing tilt due to wakefields using streak camera at CESR*, *J. Phys. Conf. Ser.* **1067** (2018) 072010.
- [8] E. Öz, F. Batsch and P. Muggli, *An accurate rb density measurement method for a plasma wakefield accelerator experiment using a novel rb reservoir*, *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.* **829** (2016) 321.
- [9] A. Caldwell et al. (AWAKE Collaboration), *Path to AWAKE: Evolution of the concept*, *Nucl. Instrum. Meth. A* **829** (2016) 3 [1511.09032].
- [10] E. Gschwendtner et al. (AWAKE Collaboration), *AWAKE, The Advanced Proton Driven Plasma Wakefield Acceleration Experiment at CERN*, *Nucl. Instrum. Meth. A* **829** (2016) 76 [1512.05498].
- [11] E. Gschwendtner et al. (AWAKE Collaboration), *The AWAKE Run 2 Programme and Beyond*, *Symmetry* **14** (2022) 1680 [2206.06040].
- [12] F. Batsch, *Setup and Characteristics of a Timing Reference Signal with sub-ps Accuracy for AWAKE*, *J. Phys. Conf. Ser.* **1596** (2020) 012006 [1911.12201].
- [13] E. Adli et al. (AWAKE Collaboration), *Experimental observation of proton bunch modulation in a plasma, at varying plasma densities*, *Phys. Rev. Lett.* **122** (2019) 054802 [1809.04478].
- [14] F. Batsch et al. (AWAKE Collaboration), *Transition between Instability and Seeded Self-Modulation of a Relativistic Particle Bunch in Plasma*, *Phys. Rev. Lett.* **126** (2021) 164802 [2012.09676].
- [15] T. Nechaeva et al. (AWAKE Collaboration), *Hosing of a long relativistic particle bunch in plasma*, 2309.03785.
- [16] V. Fedosseev et al., *Integration of a Terawatt Laser at the CERN SPS Beam for the AWAKE Experiment on Proton-Driven Plasma Wake Acceleration*, in *7th International Particle Accelerator Conference*, p. WEPMY020, 2016, DOI.
- [17] P.K. Hamamatsu, *Universal streak camera C10910-05 Catalog*, Hamamatsu, Photonics K.k (2022).
- [18] M. Wing et al. (AWAKE Collaboration), *AWAKE Status Report*, Tech. Rep. CERN-SPSC-2018-030, SPSC-SR-240, CERN, Geneva (2018).
- [19] R.C. Gonzalez and R.E. Woods, *Digital image processing*, Prentice Hall, Upper Saddle River, N.J. (2008).
- [20] A. Rosenfeld and J.L. Pfaltz, *Sequential operations in digital picture processing*, *J. ACM* **13** (1966) 471.