Detection of Type II Solar Radio Bursts

Joseph Jenkins

Swansea University
Prifysgol Abertawe

Submitted to Swansea University in fulfilment of the requirements for the Degree of Master of Research

Department of Computer Science

April, 2020

Declaration

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed ..... (candidate)

Date .....

Statement 1

This work is the result of my own independent study/investigation, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed ..... (candidate)

Date .....

Statement 2

I hereby give my consent for my work, if relevant and accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ..... (candidate)

Date .....

Abstract

Type II solar radio bursts have proven to be a useful tool for gaining insights into the behaviour of complex solar events. Some of these events currently pose a large threat to society — but if we were to gain insights into them in real-time, then their arrival times could be forecast and their damages mitigated. In this work, we processes radio sensor data pointed at the Sun to automatically detect the occurrence of type II radio bursts. We expand the scope of existing work by segmenting the signal of detected bursts, and thereby facilitating the extraction of parameters needed to gain insight into solar events. Furthermore, we detect bursts in a wavelength where no other detection algorithm currently operates.

We utilise prior knowledge of how type II bursts drift through frequencies over time to assist with the tasks of detection and segmentation. Detections are constrained to the possible physics of type II bursts by searching for regions that follow their curvature at a given frequency. The resulting high concentration of burst signal allows the use of a simple segmentation procedure based on thresholding the density of detected regions. Prior to feature extraction, we straighten out the curved regions into rectangular grids so that the resulting representations become normalised across all frequencies. The consequential reduction in variance helps to overcome the limitations of training a model when positive examples are scarce and costly to annotate. To assist with detection further, we remove low intensity background noise using a mixture of intensity and spatial analysis, and we normalise the intensity values of the sensor data using a combination of sigmoid remapping and histogram equalisation. We demonstrate the effectiveness of our methodology using the time-tested algorithms HOG and logistic regression. We evaluate our method on a custom dataset and achieve 72.5% recall, a false positive every 28 hours (69.4% precision), and 28.2% segmentation IOU. We assess the potential benefit of using specialised classifiers for different periods of solar activity but found no improvements to performance.

Acknowledgements

My deepest appreciation goes to my supervisor Adeline Paiement for the invaluable guidance they have given me. I am incredibly grateful for their continued support and patience throughout the duration of my degree — thank you for helping me to persevere.

I would like to thank collaborators Jean Aboudarham and Xavier Bonnin at the Paris Observatory for helping with the solar physics aspect of the project. I would also like to thank Xianghua Xie for being available as a co-supervisor.

I am very grateful to the James Pantyfedwen Foundation for funding my degree, as well as to my mother who has very kindly given me a place to stay. Both have allowed me to pursue my interests without the associated financial burden.

Contents

List of Tables

List of Figures

List of Acronyms

RFI Radio Frequency Interference

CME Coronal Mass Ejection

SEP Solar Energetic Particle

DH decameter-hectometric

SSN sunspot number

HE histogram equalisation

IOU Intersection Over Union

ROI region of interest

HOG Histogram of Oriented Gradients

SNR signal-to-noise ratio

LM Levenberg-Marquart

FPR false positive rate

GT ground-truth

1 Introduction

1.1 Context and motivation

Solar radio bursts are a release of electromagnetic radiation from the Sun within the radio spectrum. Since their discovery over 70 years ago [30], they have become an increasingly important topic of study due to their relation with space weather. The effects of space weather have been relatively inconsequential for most of human history, but the reliance of technological systems in modern society introduces significant risks towards the occurrence of a major event. In 1859, an infamous solar storm known as the Carrington Event resulted in a powerful Coronal Mass Ejection (CME) to strike Earth — resulting in severe disruptions to telegraph systems around the world [3]. Today, the cost of such an event in US dollars is estimated to be in the range of trillions, with the probability of another event of similar magnitude being placed at 12% per decade [32].

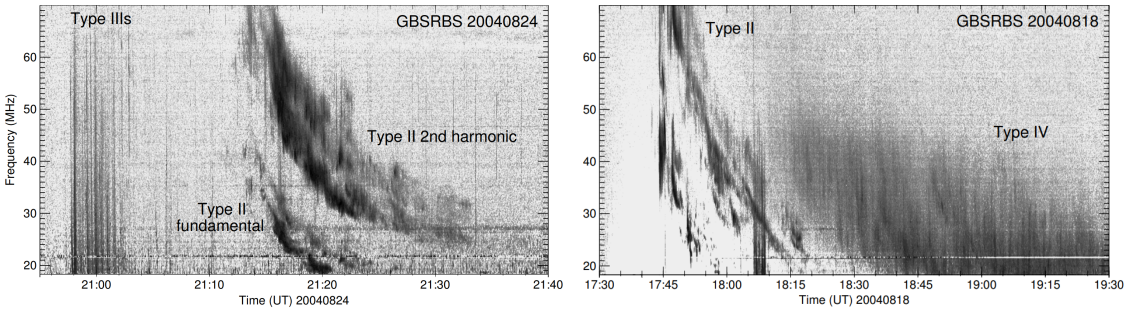
The three major components of solar storms are solar flares, Solar Energetic Particle (SEP) events, and CMEs — all of which are potential threats to society. The relation between these events and solar radio bursts is strong because they originate within the same layers of the solar atmosphere [41]. Not only does this allow radio bursts to give important insights into the behaviour of space weather, but these insights can be gained ahead of time in order to forecast the arrival times of events [25]. This is because radio bursts travel at light speed ($\sim$8 minutes to reach Earth), whereas SEP events take a few times longer, and CMEs can arrive from 15 hours to a few days after.

Radio bursts can be classified into several different types, with types II, III, & IV being the most pertinent to the application of space weather forecasting [41]. However, the only feasible way to use these bursts as a forecasting tool is to automate the process of extracting their relevant information. Several works [22, 27, 28, 29, 35] have attempted to detect the occurrence of radio bursts — however, almost all have focused on type III bursts in particular. As far as we are aware, Lobzin et al. [28] are the only ones to publish a methodology to detect type II bursts, and no current works exist for type IV bursts. Salmane et al. [35] describes a method for detecting various types of bursts although only presents results for type III bursts.

Type III bursts are characterised as having a very quick drift rate — a property that makes their spatial structure in time-frequency plots to be linear, and therefore simple to detect heuristically. Type IV bursts are also reasonably simple in structure due to their characteristic long duration broadband emissions. They may occasionally be seen drifting down frequencies, although their very slow drift means their observed frequencies remain to be relatively continuous. Type II bursts are arguably the most complex in structure. Unlike the previous types, their spatial representation in time-frequency plots becomes significantly altered in response to variances in drift rate and frequency range. Furthermore, harmonic emissions are also common which result in two (and sometimes three) distinct bands of emissions to be present at once. In addition, each band of emissions can undergo splitting which further adds to their structural complexities. Bursts of types II, III, & IV in time-frequency space can be seen in Figure 1.1.

Despite the relatively simple structure of type IV bursts, an attempt at their detection has still not been made. This is likely because they haven't been studied as much as the other two types, which may be a result of them being the rarest type. Type IV bursts are said to be a good indicator of SEP events, however, that is also true for type II bursts [17]. Moreover, a study has shown that 88% of type IV bursts are preceded by type II bursts [7], so the necessary information can likely be extracted from type II bursts instead and also at an earlier time. Because type II bursts are a strong area of research in the literature of solar physics and space weather, as well as their structure being too complex for heuristic-based detection approaches, we choose to focus our study on type II bursts.

The work of Lobzin et al. [28] used a set of 46 type II burst events and detected them with 78% recall and 88% precision (one false positive every 100 hours). This performance is considered unacceptable because around one in five events are missed, and a false positive is produced approximately once every four days. Furthermore, their detections are one-dimensional which means the amount of information that can be extracted from the events is limited. The aim of this work is to detect type II bursts with greater accuracy, as well as providing segmentations so that a wider range of burst parameters can be extracted. In addition, whereas the previous method focused on coronal type II bursts, we instead focus our approach on interplanetary type II bursts within the decameter-hectometric (DH) wavelength (1 MHz – 14 MHz). The lower frequencies of the DH wavelength represent disturbances leaving the Sun permanently, and are thus highly relevant



(a) (b)

Figure 1.1: Solar radio bursts of types II, III, & IV [41]. In both examples, a second harmonic band of emissions can be seen in the type II bursts, with all bands clearly exhibiting band-splitting. Note that the temporal scaling is different in both examples. Nevertheless, we can see that the burst in 1.1a drops down to 20 MHz in approximately half the time as the burst in 1.1b, and therefore has a quicker drift rate. In 1.1a, we see a group of type III bursts appearing as straight lines in the plot. In 1.1b, we see a type IV burst occupying a wide frequency range, and in this particular instance can be seen showcasing a slow drift.

for space weather [17]. DH type II bursts are caused by shock-accelerated electrons driven by CMEs, and hence provide useful information regarding the corresponding shocks and CMEs [18]. Nevertheless, type II bursts at all wavelengths are useful to study, and combining both detections could be used to facilitate multi-wavelength studies.

1.2 Contributions

- **Automated detection and segmentation of type II solar radio bursts** — As far as we are aware, we are the first to supplement radio burst detections with segmentations for greater characterisation ability. We also believe we are the first to detect type II bursts within the DH wavelength.
- **Integration of prior physics knowledge** — Expanding on earlier work, we utilise the known drift model of type II bursts to simplify the task of detection and segmentation. The model is used to describe the expected trajectory of type II bursts within time-frequency space.
  - **Normalisation of signal orientation across all frequencies** — By integrating the knowledge of how frequency relates to the orientation of signal, we reduce the variance associated with this effect from our resulting features. This greatly simplifies the task of learning a predictive model, and also makes the model more adaptable to new instruments with different frequency ranges.
  - **Preservation of information and spatial context through the application of a two-dimensional coordinate transform** — We improve on the previous approach by utilising both the frequency and time dimensions during transformation. The previous approach considered the frequency dimension only, resulting in information to be lost from only partially describing the structure of bursts. Our approach resolves the issue of information loss by ensuring that the entire structure of bursts within time-frequency space is accounted for. In addition, our approach ensures spatial context is preserved through the use of a one-to-one sampling rate between the pixels within the original and transformed coordinate spaces.
  - **Constrained search** — Our detector operates within our transformed coordinate space, and as a result its search path becomes tied to the drift trajectory of type II bursts. Within image space, this corresponds to a curved ROI whose curvature is a function of frequency. This ensures that any detections are constrained to the possible physics of type II bursts, since the detector looks for specific shapes depending on the frequency. Furthermore, traditional approaches to combat variances in scale are inherently designed for natural images, and hence do not translate well to time-frequency

space; the drift trajectory of type II bursts would be in violation of the drift model, resulting in shapes that are impossible within the context of the real-world. Our transformed coordinate space means that any transformations directly correspond to the curvature of the drift trajectory, and therefore any scaling becomes locked within the possible physics of the drift model.

- **Preparation of dynamic spectra for feature extraction** — We facilitate the task of pattern recognition by remapping the unbounded intensity range of the raw sensor data to a fixed range. We believe our approach to be transferable to different instruments, since little to no assumptions of the intensity distribution are needed.
- **Pixel-wise annotated dataset** — We hand-annotate a varied selection of 283 type II burst events for segmentation training and evaluation.

1.3 Thesis overview

- **Chapter 2: Background** — We begin by investigating works that have integrated domain knowledge to assist with the task of radio burst detection, and then we look more closely at the existing methodology used to detect type II bursts. We then review techniques that are relevant for developing a computer vision pipeline: histogram equalisation for contrast enhancement, Histogram of Oriented Gradients for feature extraction, and supervised machine learning for classification.
- **Chapter 3: Data** — We present our dataset, including the process of collection and annotation. We also identify some challenges of the data, including issues that relate to the instrument, radio bursts, and the Sun.
- **Chapter 4: Methodology** — This section is split into two main subsections: data preprocessing and burst localisation. The preprocessing section mainly focuses on removing the radio background noise as well as normalising the intensity values of the raw sensor data. The localisation section presents the methodology for detecting and segmenting type II bursts. We begin this section by presenting the design of an ROI modelled after the curvature of type II bursts. A methodology is then devised for generating training samples from our annotations, where optimisation techniques are used to select a finite number of ROI parameters that best describe our annotations. We then describe our detection pipeline for detecting burst segments, followed by our approach to post-processing detection ROIs to produce segmentations.
- **Chapter 5: Experiments** — We begin by outlining the dataset used for training, and then present results for experiments based on the stages of our methodology: preprocessing, feature extraction, detection, and segmentation. The preceding hierarchy is used for evaluating optimal parameter configurations. Failure-case analyses are carried out to identify areas for improvement. We also test the effectiveness of utilising distinct classifiers for different periods of solar activity.
- **Chapter 6: Conclusion** — This section summarises our methodology and outlines its suitability for application to other instruments. We identify areas of future work such as using the drift model to group burst segments and improve detection performance through context-based reinforcement.

2 Background

2.1 Solar radio bursts

2.1.1 Drift rate of type II bursts

Solar radio bursts are typically analysed as two-dimensional plots of frequency over time, or 'dynamic spectra' as known within the solar physics community. Within this data representation, the 'structure' of radio bursts are a consequence of how they drift through frequencies over time. The nature of how type II bursts drift through time — and hence its resulting structure — can be modelled by a power law which describes the relationship between frequency and drift rate in MHz/s [2]

$$-df/dt = \alpha f^{\psi}, \tag{2.1}$$

where α and ψ are a scaling factor and power index on the frequency f, respectively.

Given the model, we know that a burst will always decrease in frequency over time, and that the rate at which it does so also decreases with frequency. The effect of this can be clearly seen in Figure 1.1a, where the characteristic curvature of type II bursts is a result of the continuously decreasing drift rate. Assuming that α and ψ are known and remain static over a burst's lifetime, then the curvature of the burst also becomes known ahead of time. In practice, the exact values of α and ψ cannot be known as a prior, but an expected range of values would sufficiently describe a range of possible drift trajectories. Using a collection of type II burst events across various wavelengths, Aguilar-Rodriguez et al. [2] plotted frequency against drift rate to obtain the best fitting power index for each wavelength. Presumably, the plot could also be used to determine an upper and lower boundary of scaling factors and power indices.

Lobzin et al. [26] found that $\psi \in [0.6, 1.3]$ for eight coronal type II bursts within 25 – 180 MHz. As $\psi \approx 1$, they state that re-mapping the frequency (f) coordinates as $1/f$ results in the curvature of type II bursts to be reduced, as can be seen in Figure 2.1. Lobzin et al. [28] later used this technique to reduce the difficult problem of burst detection down to the simple task of recognising straight lines (see Section 2.1.3 for details).

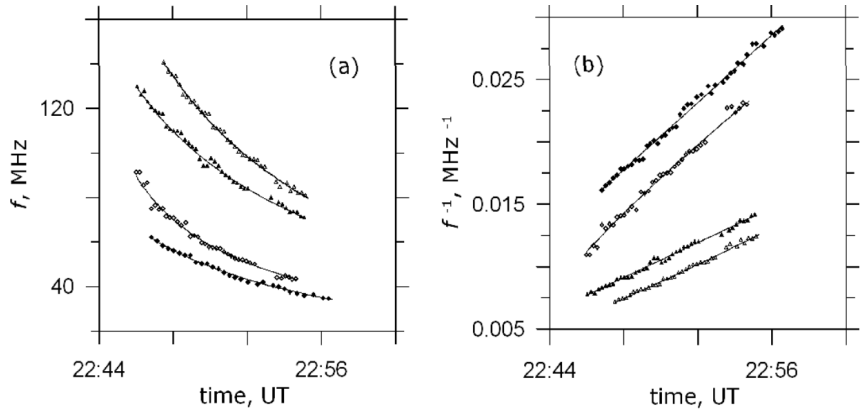
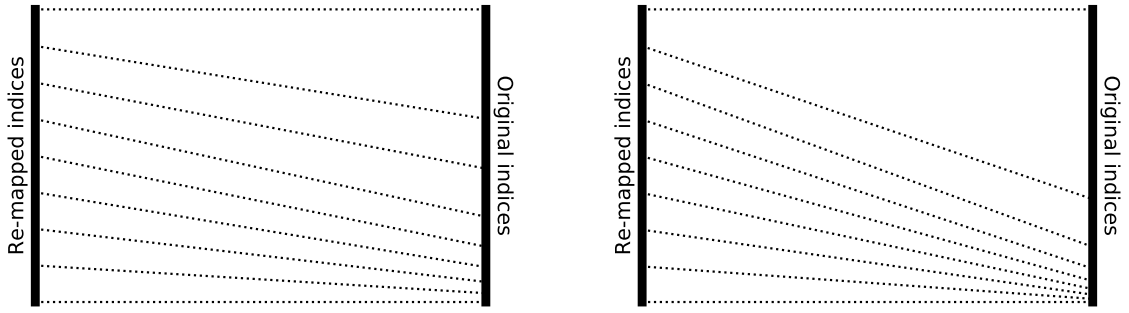


Figure 2.1: Re-mapping f to $1/f$ to straighten the curvature of type II bursts [26]. a) Before (f), and b) after ($1/f$).

While this technique does help to reduce the curvature of bursts, a major issue is the fact that it only considers the frequency dimension for re-mapping. The procedure arranges the frequencies in such a way that the curvature of bursts are forced into a linearised representation. Frequencies that do not correspond to an increase in time are consequently discarded, and frequencies that persist over several temporal samples correspond to the sampling of singular data points many times. The transformation being unable to capture the full structure of bursts in time-frequency space ultimately results in a loss of information.

The increased drift rate at higher frequencies means that several frequencies are likely to occupy the same temporal sample, and will therefore be the predominant source of information loss. The instrumentation design of Learmonth, the data used in [28], is arguably well suited to counteract this effect; frequencies are split into two bands each with 401 channels: 180 – 75 MHz and 75 – 25 MHz, corresponding to a resolution ratio of 1:2.1 for the two bands. The decreased resolution at

the higher frequencies helps to prevent too many frequencies occupying the same temporal sample, and therefore helps to reduce the amount of information loss. Nevertheless, the $1/f$ re-mapping still results in 33.5% of information to be lost. For Wind/WAVES, the instrument used in this study, information loss is substantially greater at 56.3%. Type II bursts within Wind/WAVES also occur mostly at the higher frequencies, resulting in the effective information loss to be even more significant. Figure 2.2 shows the effect of the re-mapping for both Learmonth and Wind/WAVES. It is clear that while the procedure may be somewhat acceptable for Learmonth data, the degree of information loss within Wind/WAVES is far too great to be practical.



(a) Learmonth (b) Wind/WAVES

Figure 2.2: Effect of $1/f$ re-mapping on frequency indices for a) Learmonth, and b) Wind/WAVES. In both cases, we can see that the higher frequencies are compressed to a smaller range, and the lower frequencies are up-scaled to a larger range. The effect of compression and up-scaling is much more significant for Wind/WAVES; about half of the upper frequencies are compressed down to a very small range, and conversely a very small range of low frequencies are up-scaled to occupy about half of the data.

2.1.2 Galactic background noise

In the early 1930s, Karl Jansky established the field of radio astronomy when an astronomical radio source had been observed for the first time [20]. The observed signal peaked every $\sim$24 hours, which led to the discovery that it was coming from a fixed source: the constellation of Sagittarius. This mysterious radio source was later designated Sagittarius A in the 1950s, and then in 1974 a bright and compact component of the source was identified and named Sagittarius A*. Sagittarius A* is now well known to be a supermassive black hole in the centre of the Milky Way galaxy [1].

It is no coincidence that the signal from Sagittarius A became the first observation of an astronomical radio source; its abundance within radio observations certainly made it a prime candidate. The property of being abundant does, however, interfere with the objective of observing other astronomical radio sources. Figure 2.3 shows an example of how this noise is present within observations of Jupiter, and how the signature of the noise varies with frequency and time. The two peaks in 1972 and 1984 correspond to the orbital period of Jupiter intersecting with the line of sight between Earth and the centre of the galaxy. The presence of this noise does have an advantage, however: a reliably occurring source that's well understood serves as an excellent reference for calibration of radio telescopes [12]. Still, though, the presence of background noise can make it difficult to identify weak signals comparable to the strength of the background, and can also hinder the application of data processing techniques for feature enhancement and pattern recognition.

To ease the process of detecting solar radio bursts, Salmane et al. [35] removed the background noise by making use of the fact that the distribution of its intensities is Gaussian. For each temporal step, they used 7 hours of previous data to compute the mean and standard deviation for each frequency channel. They argue that over a long enough period, the computed parameters will be representative of the background, and hence can be used to subtract away the noisy signal. They also argue that this approach can also be used to remove RFI (Radio Frequency Interference), which is another type of noise where intense signal spans a frequency channel to create horizontal structures within the data. Figure 2.4 shows their example of subtracting the background using the mean intensity as the threshold, corresponding to a theoretical reduction of 50% of the background. Their example does not clearly show the effectiveness of removing the background, but we do see

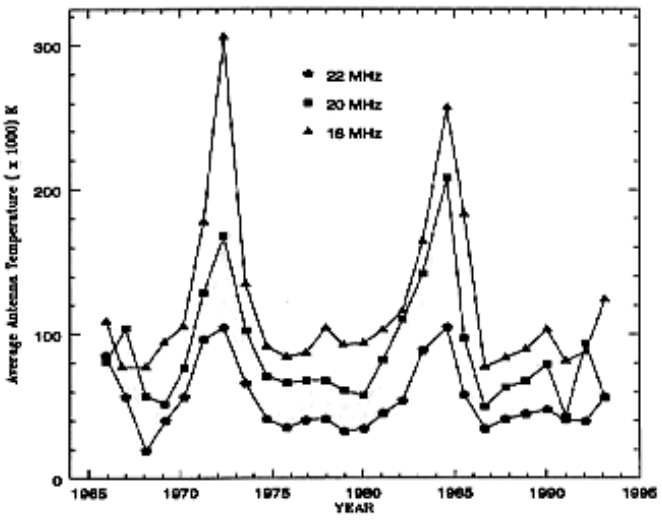


Figure 2.3: Temperature of galactic background over 30 years of Jupiter observations [14]. X=Year{1965,1970,1975,1980,1985,1990,1995}, Y=Average antenna temperature (x1000) K{0,100,200,300}; Series={18,20,22} MHz.

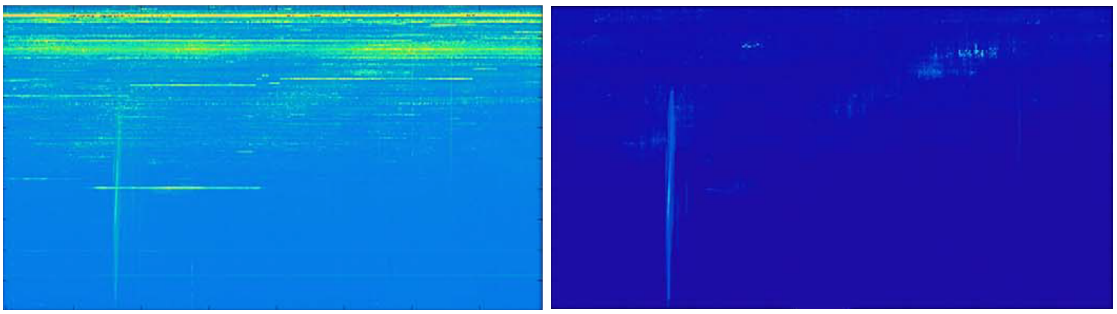


Figure 2.4: Before and after background subtraction (Salmane et al. [35]).

a clear reduction of RFI at the top of the image.

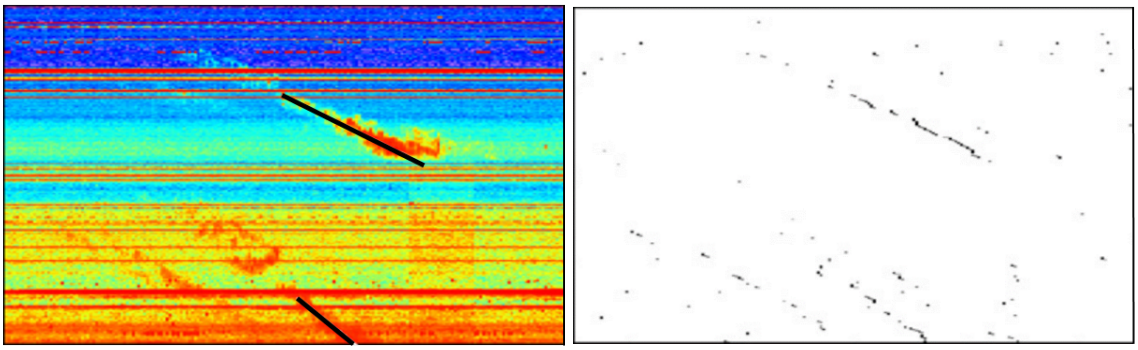
An issue with the approach used by Salmane et al. [35], at least for the data used in our own study, is that the intensity levels of the background are many orders of magnitude smaller than the possible range of the sensor. As a consequence, the mean as a metric has the tendency to significantly overestimate the true intensity of the background due to the inclusion of outliers. For example, the average intensity of background signal within our data is $\sim$1.05, and a strong type III burst may have an intensity of 50+. Even just a single occurrence of a 3 minute type III burst over the course of 7 hours would skew the mean from $\sim$1.05 to $\sim$1.4, which would be enough to completely remove many instances of type II bursts. Furthermore, the temporary presence of RFI means that its intensity will often be underestimated, resulting in a reduction of intensity as opposed to a complete removal of the noise. Some cases of this happening can be seen in Figure 2.4. When designing a detector, these inconsistencies could potentially be more of a detriment rather than an advantage.

2.1.3 Detection of type II bursts

As far as we are aware, Lobzin et al. [28] is currently the only existing method for detecting type II bursts. The fundamental aspect of their methodology is based around reducing the curvature of type II bursts using the $1/f$ remapping described in Section 2.1.1, and then using a Hough transform to detect straight lines. Using a selection of 46 events over 510 hours worth of data, their method detected 36 events and 5 false positives. They evaluated their method using data from the Learmonth solar radio spectrograph which covers the frequency range 25 – 180 MHz. The data has been conveniently quantised into unsigned bytes but details on this process are inaccessible.

Prior to detection, they preprocess the data with the aim of producing a binary image that

separates burst signal from non-burst signal. They begin by discarding the 25 – 44 MHz frequency range, stating that the prevalence of interference at this range is very high and will lead to an increase in false positives. For the remaining interference, they report that their signal corresponds to higher intensity values relative to other signal such as type II bursts. Therefore, they use histogram equalisation to increase the dynamic range of burst signals. They then process the data using a median filter, followed by a binarisation of the image. The criteria they use for binarisation is to only preserve signal when it is the maximum intensity within a 3 pixel temporal window. Morphological thinning is then applied to reduce the thickness of morphological structures down to a one-dimensional line. To reduce the amount of non–type II signal, they remove any isolated pixels as well as remove any morphological structures that are seen to increase with time. Figure 2.5b shows an example of a complete preprocessed image, where Figure 2.5a shows the resulting detection.



(a) (b)

Figure 2.5: Detection of type II bursts (Lobzin et al. [28]). The black lines in 2.5a show the detections of type II bursts after using a Hough transform on the processed image 2.5b.

After applying the Hough transform, they group together segments separated by a short gap as being part of the same event. Segments that overlap in time are also grouped, since it is assumed that they correspond to harmonic emissions. To maximise the trade-off between true and false positives, they use their selection of events to optimise the parameters of the Hough transform. They state that a weakness of their approach is that short segments are difficult to recognise, since the ability to detect them would correspond to a substantial increase in the false positive rate. We can clearly see from Figure 2.5a how this fact, combined with the aggressive preprocessing, results in a large portion of burst signal to remain undetected. Visual inspection of type II bursts within Wind/WAVES, the instrument used in our own study, has shown that the signal of type II bursts often represents short-lengthed structures. Preliminary experiments for Wind/WAVES resulted in 10% recall and a 10% false positive rate [4].

2.2 Histogram equalisation

Histogram equalisation is a technique used to spread out the distribution of intensity values that are otherwise concentrated within a small range. The aim of this procedure is to maximise the potential contrast available within a discrete number of grey levels L. To achieve this, we can consider a transformation function T which maps the intensity values r of an image to its new values s. Assuming r and s are normalised to $[0, 1]$, then we can consider their histograms to represent probability density functions in which the intensities act as continuous random variables. The definition of T thus becomes a problem of satisfying

$$T(p_r(r)) = p_s(s), \tag{2.2}$$

where p_r and p_s are probability density functions and p_s is uniformly distributed. A transformation that satisfies this condition is derived in [15], which is given as

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} \frac{n_j}{n}, \quad 0 \leq k \leq L-1, \tag{2.3}$$

where k is a grey level, n is the total number of pixels in an image, and n_j is the total number of pixels that have a grey level r_j. The output of $T(r_k)$ is thus the cumulative probability of a given grey level, which is then used as the intensity value for the new image.

2.3 Histogram of Oriented Gradients

In [10], Dalal & Triggs proposed a method for detecting humans within images. They note that human detection is a challenging task due to their wide variations in appearance and pose, and is further complicated by variations in illumination and background clutter. Their proposed solution is to use locally normalised histograms of oriented gradient (HOG) descriptors, arguing that local shape information is well described by the distribution of local intensity gradients.

Images are divided into smaller regions known as 'cells', with each cell being used to compute a HOG descriptor. To tackle the issue of illumination variance, they group cells into larger spatial regions known as 'blocks', where each block is used to normalise the contrast of the cells within them. Each block is then combined into a single feature vector which can be used for classification. To improve the process of normalisation, which they demonstrate to be a critical component for achieving good performance, they choose compute HOG descriptors on overlapping blocks. Despite the seemingly redundant nature of duplicating cells within the final feature vector, they show that the resulting increased density of block normalisation greatly improves performance.

Compared to the state-of-the-art at the time, the use of HOG descriptors greatly improved results. HOG descriptors cue mainly on silhouette contours, so they note that as long as a human is roughly in an upright position, then this method performs well even when limbs and body segments change appearance and location. The method's ability to perform well against a difficult task with wide variations makes it a well-suited method for many other object detection tasks. For this reason, HOG descriptors still remain in use as an effective and efficient approach for extracting features [9, 31, 40].

2.4 Supervised machine learning

The aim of machine learning is to learn the underlying model that describes a dataset so that new predictions can be made. Usually, we will know the target values for a set of samples within a dataset, where this knowledge can be used to guide the process of learning. This is done by validating that the hypothesised model is able to describe the data, and is known as *supervised* learning. Without knowing the target values, then we must look for underlying patterns within the data without the help of validation, and is known as *unsupervised* learning. This section focuses on the supervised variant.

The most basic form of supervised learning is linear regression, which is a statistical method used to predict continuous variables given an input of features. Given a line fitted to a set of data points, a target variable $\hat{y}$ can be predicted from an input feature x using the equation of a line

$$\hat{y} = c + mx, \tag{2.4}$$

where c is the y-intercept and m is the slope of the line. More generally, this can be applied to an arbitrary number of input features by making predictions from an n-dimensional line

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n, \tag{2.5}$$

where $h_\theta(x)$ (replacing $\hat{y}$) is a hypothesis for a feature vector x, θ_i (replacing m) is the weight of feature x_i, θ_0 (replacing c) is the bias term, and n is the number of features. This can be simplified to

$$h_\theta(x) = \theta^T x, \tag{2.6}$$

where $x_0 = 1$.

Figure 2.6 shows an example of linear regression under the simple two-dimensional case. The residuals can be used to evaluate how well the fitted line describes the observed data points. For example, we could use the mean squared error

$$J(\theta) = \frac{1}{2m} \sum_{j=1}^{m} (h_\theta(x^{(j)}) - y^{(j)})^2 \tag{2.7}$$

where m is the number of data points, to define a *cost function* J that quantifies the error between a hypothesis $h_\theta(x)$ and the actual data points. The process of *training* a model such that new predictions can be made thus becomes a problem of finding θ that minimises $J(\theta)$. Gradient descent is an algorithm which attempts to solve this problem by computing the gradient of $J(\theta)$ and updating θ in the direction that follows the path of steepest descent

$$\theta_i := \theta_i - \alpha \frac{\partial}{\partial \theta_i} J(\theta), \tag{2.8}$$

where α is the *learning rate* which controls the step size of each iteration. The algorithm stops when $J(\theta)$ converges to a local minimum or after a certain number of iterations has been reached. The use of gradient descent influences the use of the $\frac{1}{2m}$ term within the cost function; averaging the error across all data points allows α to be robust against changes to m, and adding the 2 helps to simplify the derivative.

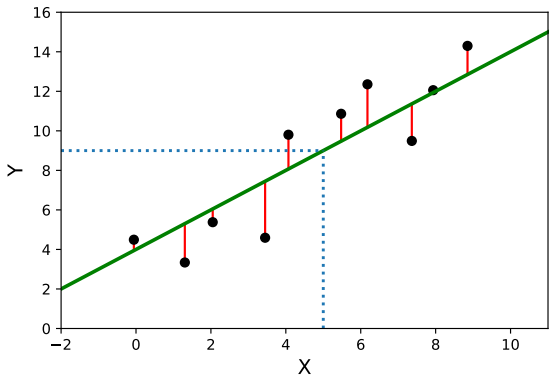


Figure 2.6: Linear regression. The green line shows the line of best fit given the observed data points. In this example, $c = 4$ and $m = 1$. The fitted line is used to predict that $y = 9$ when $x = 5$. The differences between the data points and the fitted line (shown in red) are known as *residuals*.

Linear regression works well when our target predictions are continuous variables, but prediction tasks often take the form of categorising features into discrete outcomes, also known as *classification*. In the case of binary classification, a simple workaround could be to assign one category to the value 0, and the other to the value 1. Then, we can fit a line to the data using linear regression and use the centre value 0.5 as a threshold point to partition the data into two classes. The partition used for classification is also known as the *decision boundary*. However, we can see from Figure 2.7a that this solution does not work too well. Hypotheses can correspond to values below 0 or above 1, and outliers consequently have a large impact on the decision boundary, even when it is clear that the decision boundary should not be changed.

To overcome these shortcomings, the sigmoid function

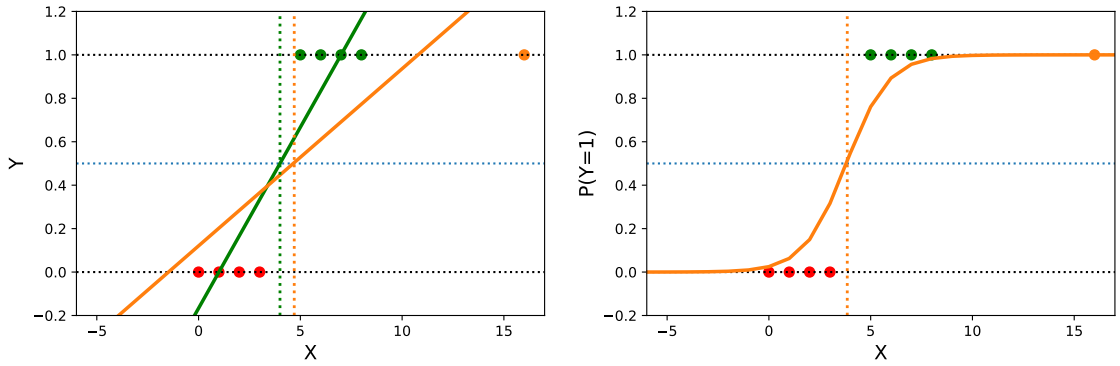
$$g(z) = \frac{1}{1 + e^{-z}} \tag{2.9}$$

is used to transform the range of hypotheses

$$h_\theta(x) = g(\theta^T x) \tag{2.10}$$

such that $0 \leq h_\theta(x) \leq 1$. This method is known as logistic regression, which is an adaptation to linear regression for solving classification problems. A nice property of logistic regression is that a hypothesis $h_\theta(x)$ becomes equivalent to the probability that $y = 1$. As in Figure 2.7b, we can use this for classification by setting a threshold at 0.5 so that the most likely outcome is predicted. Alternatively, the probability output could be used as a single component within a broader decision-making model.

As with linear regression, we can train a model to make new predictions by using gradient descent to find θ that minimises $J(\theta)$. However, the use of the non-linear sigmoid transformation results in the cost function from Equation 2.7 to be non-convex. In other words, the function has many local minima and will cause gradient descent to converge to a point other than the global



(a) Linear regression

(b) Logistic regression

Figure 2.7: Linear regression versus logistic regression for classification. The central blue line is a threshold at $y = 0.5$, where a decision boundary is modelled at the intersection of the fit. a) In linear regression, the addition of the orange outlier drastically affects the slope of the fit. b) In logistic regression, the decision boundary with and without the addition of an outlier remains unchanged.

minimum. To solve this, the following cost function known as log loss (or cross-entropy loss) is used instead

$$J(\theta) = -\frac{1}{m}\sum_{j=1}^{m} y^{(j)} \log h_\theta(x^{(j)}) + (1 - y^{(j)}) \log(1 - h_\theta(x^{(j)})). \tag{2.11}$$

The purpose of log loss is to evaluate how well the estimated probabilities match the actual class labels. We can see that the function is composed of two terms: one multiplied by $y^{(j)}$, and the other multiplied by $1 - y^{(j)}$, where $y^{(j)}$ is the class label for sample j. Thus, depending on the class label, only one of the two terms are used to contribute to the error. In each case, by taking the negative log of the probabilities, the penalty of the cost function increases exponentially as the estimations diverge from the class labels. This property means that the function will be convex, and hence allows the use of gradient descent to find a global minimum.

2.5 Summary

We have reviewed the relevant literature needed to support the design and development of a computer vision pipeline, specifically for the task of detecting type II solar radio bursts. We investigated what is currently the sole methodology used to solve the problem of detecting type II solar radio bursts. Although we cannot directly compare the results between the evaluated method and our own method due to the difference in datasets (and notably, the difference in wavelength domain), the existing methodology serves as a valuable source of known pitfalls and potential solutions specific to the detection of type II bursts. We also investigated works that have made use of prior knowledge to aid in the task of detecting solar radio bursts. Specifically, we looked at how the known drift model of type II bursts has been used to normalise their appearance, and how the known properties of the background has been used to help remove it from the data. Given that the effective use of data-driven models is impeded by the limited access to real-world examples, we aim to utilise and expand the existing ideas of using prior knowledge to complement the use of data-driven models.

In order to make use of data-driven models, we investigated techniques that are able to describe image data with meaningful features. We reviewed the Histogram of Oriented Gradients (HOG) algorithm, which was originally applied to the task of detecting humans within images. Given the geometrical nature of radio bursts in 2D space, we find it appropriate to utilise a shape descriptor for extracting relevant features. It was noted how the method was robust to variances in illumination, background clutter, and body/limb appearances. This transfers well to the task of radio burst detection, since we need to be robust against changes in intensity levels, overlaps with irrelevant signal, thickness and length of the burst, as well as any other complexities such as band-splitting. It was noted that the method performed well as long as a human was roughly in an upright position. Thus, similar to the work of Lobzin et al. [28], we aim to use prior knowledge

of the type II burst drift model to normalise the orientation of burst signal, thereby enabling the extraction of consistent HOG features.

Popular algorithms such as HOG already have existing implementations that are well tested, very efficient, and accurate to the original methodology described. However, the implementation of these algorithms are designed to work with images, and hence expect the input to take the form of unsigned bytes. Therefore, to capitalise on the existing efforts of the computer vision community, we must first normalise the intensity values of our data to be in the fixed range of $0 - 255$. One caveat to this requirement is the fact that the range of intensity values within our data is very wide, so we must normalise the intensity range in such a way that preserves the useful signal while maintaining a high degree of contrast. We choose to use the sigmoid function to transform the raw intensity values into a fixed range, allowing extreme values to be clipped to an upper boundary while the contrast of the usable data is stretched. However, due to the sigmoid function operating on individual values, we cannot guarantee that this procedure will result in a suitable normalisation of contrast for all data samples. Therefore, we have reviewed an adaptive procedure, histogram equalisation, which utilises a histogram analysis to spread out the intensity values across the usable range of values.

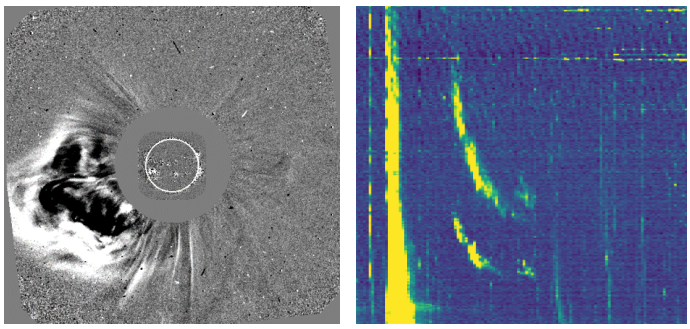
Finally, we reviewed the use of supervised machine learning algorithms to create data-driven models. We introduced the logistic regression algorithm, a popular and simple statistical method used to classify features into binary outcomes (positive or negative). We will use logistic regression to model a decision boundary of HOG features by training the model on a selection of real-world events. We have looked at how cost functions can be defined and minimised to optimise the generalisability of parameters. Since the process of selecting events will require a mapping from pixel-wise annotations into discrete-valued parameters, knowledge of using cost functions will help us to define parameters that generalise well to our dataset. Once we have a model that is able to effectively classify between burst and non-burst signal, we will apply the classifier to a sliding window–based detector to extract tightly localised regions that contain burst signal. We will later take advantage of the probabilistic nature of the model by utilising the continuous predictions within a segmentation-based procedure.

3 Data

In this study, we use data observed by the WAVES [5] instrument aboard the Wind spacecraft, which has been collecting data since its launch in 1994. The instrument consists of two bands of radio receivers: RAD1 operating at 20 – 1,040 kHz, and RAD2 operating at 1.075 – 13.825 MHz. Each band consists of 256 linearly spaced channels. It is very uncommon (∼3% of cases) for type II bursts to be solely visible within the extremely low frequencies of RAD1, so we exclusively use RAD2 in this study.

3.1 Collection and annotation

We source our data from NASA's public archive[1], where the data has been averaged into minute-long samples. Their event catalogue[2] of DH type II bursts is used to collect positive examples. The catalogue contains a collection of 511 events from 1997 – 2016 which have been found through so-called "radio-loud" CMEs. The CMEs are called radio-loud due to their ability to produce type II bursts, which can be seen in Figure 3.1.



(a) SOHO (b) Wind/WAVES

Figure 3.1: Example of a radio-loud CME (2013/07/04 21:00). a) Observation of the sun using a coronagraph. The white circle represents an outline of the sun, and the surrounding grey disk is used to block the sun's glare. A CME is seen being ejected from the sun. b) The produced type II burst as a result of the CME in 3.1a.

We build a catalogue of negative samples by considering two sub-classes: type III bursts and 'background'. We sample type III bursts explicitly because their intense presence has the potential to be a common source of false positives. The background sub-class aims to capture the general day-to-day features of the data as well as any other prominent events. We use a catalogue[3] of automated detections for the type III bursts, and we consider any event outside of the type II and type III burst catalogues to be the background. Because we consider type III bursts from an automated catalogue, there is the risk of sampling data that does not actually contain a type III burst. We consider this to be acceptable because we are only concerned about the detection of type II bursts. In fact, by including samples that have proven to fool an existing automated model, then their inclusion may help to prevent our own model from suffering from the same shortcomings.

We use the reported starting times from each of our three catalogues (type II bursts, type III bursts, and background) to create three-hour windows: 15 minutes before the event and 2:45 hours after. We then filter out any negative windows such that none of our events overlap with each other. We annotate a selection of 283 type II windows by overlaying the bursts with a pixel mask. Harmonics of the burst are annotated separately (see Figure 3.2 for the possible annotations). The general quality of annotations have been verified by an expert, although it is difficult to guarantee the correctness of fine-grained details such as annotation at pixel level and harmonic classification. Further details on the dataset used to train and test our model can be found in Section 5.1, and a list of events used can be found in Appendix 1.

[1] `https://cdaweb.gsfc.nasa.gov/pub/data/wind/waves/wav_h1/`
[2] `https://cdaw.gsfc.nasa.gov/CME_list/radio/waves_type2.html`
[3] `ftp://ftpbass2000.obspm.fr/pub/helio/hfc/obsparis/frc/rabat3/`

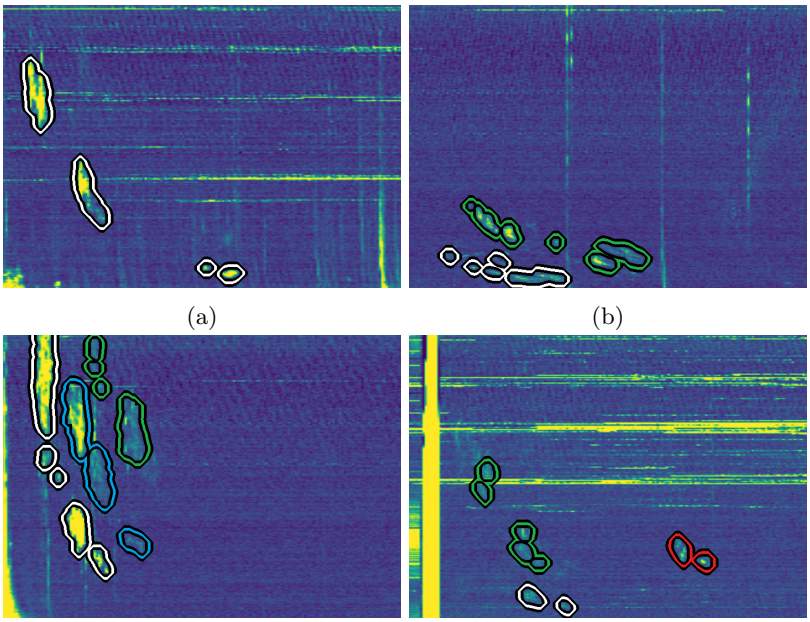


Figure 3.2: Type II burst annotations. a) One harmonic (white). b) Two harmonics (+green). 3) Possibly three harmonics (+blue). d) Possibly two distinct events (+red).

3.2 Challenges

3.2.1 Data properties

One of the challenges faced with processing scientific data is the presence of information that is either noisy or of low contrast [24]. This can be an issue for many feature extraction techniques such as those relying on analysing intensity gradients; noisy information may produce strong gradient responses, whereas useful information may not. Radio astronomy in particular is no stranger to this phenomenon [42]. Figure 3.3 shows how man-made interference can severely corrupt the signal of certain frequencies observed by Wind/WAVES. In addition, the galaxy produces a constant stream of random Gaussian noise that defines the background of our data. This effect can be seen throughout all of our examples (e.g. Figure 3.2). While the intensity distribution of the background is clearly defined, the arbitrary signal strength of other emissions can make the data more difficult to process. Figure 3.4 shows two extremes of signal strength for type II bursts. Note that for visualisation purposes, we clip the intensity values in our data to 1.3. For bursts that are particularly strong, this can give the appearance of having homogeneous intensity values.

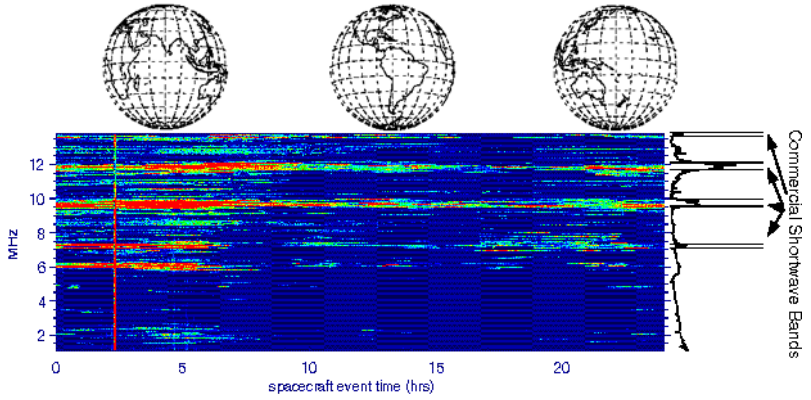
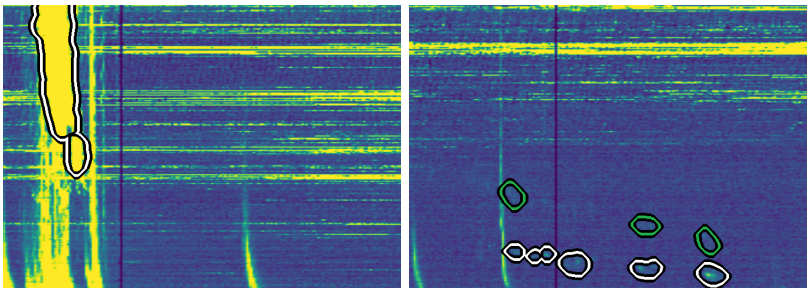


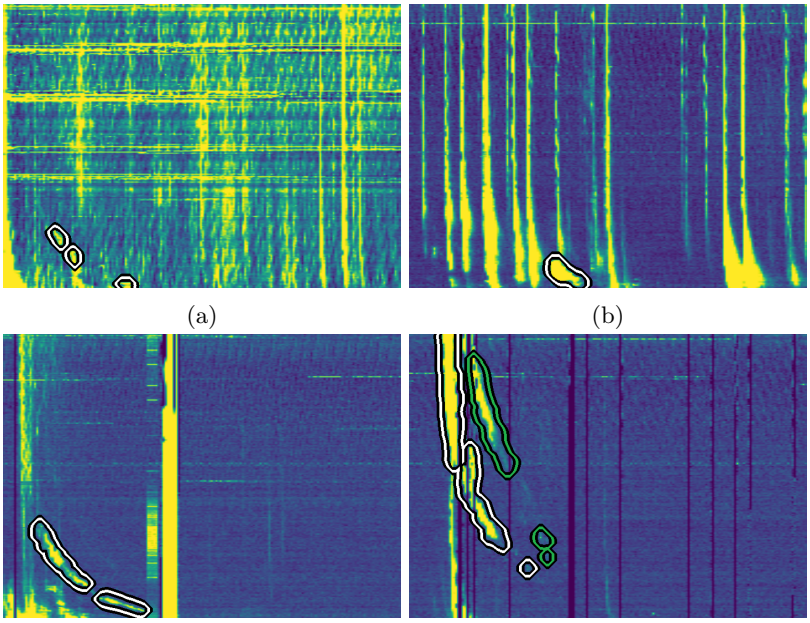
Figure 3.3: Radio Frequency Interference (RFI) from Earth [23]. The strong horizontal bands (RFI) correspond to the frequency bands used by national radio stations. The severity of corruption changes throughout the day depending on what part of the world faces the instrument.



(a) (b)

Figure 3.4: Bursts of varying signal strength. a) A strong burst (1.3+ intensity). b) A weak burst merged with the background (particularly in the case of the second harmonic).

Aside from the difficulties of general radio astronomy, the specifics of instrumentation and its task (in this case: using Wind/WAVES to observe radio emissions from the sun) come with its own unique challenges. The instrument itself requires daily calibration that produces a large block of high intensity signal (Figure 3.5c), and sometimes observations can be missed (Figure 3.5d). Both calibration signals and missing data correspond to sharp changes in intensity, so we must make sure our detector is robust to these issues. Fortunately, missing observations only account for ∼2.3% of samples, and it is rare for consecutive samples to be affected (Figure 3.6). Therefore, it is unlikely that type II bursts will be significantly occluded.



(a) (b)

(c) (d)

Figure 3.5: Some examples of non–type II signal. a) The usual low intensity background has been overpowered by a type III storm. b) A type II burst is seen amidst a cluster of type III bursts. c) Calibration of the instrument results in high intensity signal to span the entire frequency range. d) Periods of missing data usually affect the entire frequency range and can sometimes affect consecutive observations.

When observing the sun, the emissions of type II bursts are one of many classes of emissions that can be produced. Non–type II emissions could potentially cause issues with misclassification: either as false positives, through the emissions having type II–like features; or false negatives, through the emissions occluding positive events. Figures 3.5a & 3.5b show examples that may cause difficulty with misclassification. In the event that bursts are still detected successfully, distinguishing the boundary between the type II signal and the overlapping non–type II signal still remains a challenge for segmentation.

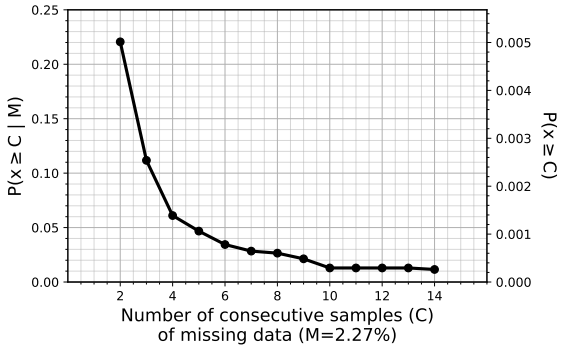
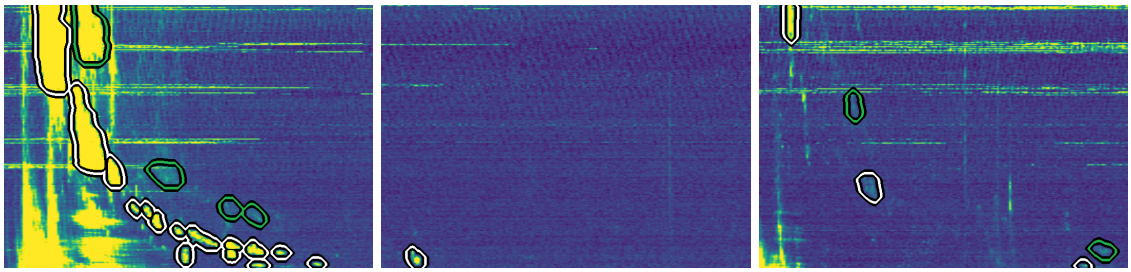


Figure 3.6: Probability of consecutive (temporal) samples being affected by missing data. We use our type II windows to estimate the probability of a given sample x belonging to a chain of at least C missing samples. Left) Probability given we know x is missing. Right) Probability for a random x.

The signal strength of various emissions can vary widely between events. Signal of type II bursts are usually weaker than other bursts and RFI, but there are many exceptions where the opposite is true. The strength of type II signal may also vary within the same event. A burst may have sporadic rises in intensity (Figure 3.7c), and harmonics are often weaker than their lower order counterparts (e.g. Figures 3.2c & 3.4b). As before, there are many exceptions (Figure 3.2b). Furthermore, as lower frequency emissions correspond to signal further away from the Sun, the intensity of type II bursts will naturally decrease over time [16]. Because we are using data from the lower end of the radio spectrum, this often results in the intensity to drop below the background. The variances of intensity in both relevant and irrelevant signal could make it difficult to classify the bursts effectively using intensity-based features. In an attempt to prevent issues with data quality impeding the quality of our detector, we consider the use of preprocessing techniques to alleviate the issues.

When plotting a burst as a spectrogram, its shape is a consequence of how it drifts through frequencies over time; it is shown in Section 2.1.1 that drift rate — and therefore a burst's curvature — is dependent on frequency. In general, the curvature will be vertical at the higher frequencies, curved around the mid-to-lower frequencies, and then horizontal at the lower frequencies. This effect can be seen throughout all of our examples such as in Figure 3.4. However, the specifics of the frequency-drift dependency vary widely between events and is a factor of initial drift rate and harmonic order. Its state may also suddenly change in response to changes of state in the sun. The starting frequency and duration of a burst will also significantly affect its overall shape, and even bursts that share the same properties can present large differences in appearance. Figures 3.7a & 3.7b show the extreme differences in appearance due to differences in starting frequency. Figure 3.7c shows that even when a burst is large, its appearance may be more similar to the smaller bursts due to the visible portions of the burst being sporadic. To get around these issues, we consider detecting bursts in segments as opposed to the full event all at once. However, this introduces a new challenge of reconstructing the full event from the smaller segments. To simplify the tasks of detection and reconstruction, we also consider normalising the shape of bursts with respect to the frequency-drift dependency.



(a) (b) (c)

Figure 3.7: Different sizes of bursts. a) The burst spans the entire frequency range and most of the temporal window. b) The burst covers a very narrow frequency and temporal range. Its low starting frequency means that most of its signal is outside of the visible frequency range. c) A large burst, but most of its signal has faded into the background with intermittent periods of higher intensity.

3.2.2 Variability of solar activity

The sun is a complex, dynamic entity that changes in behaviour over time; its level of activity periodically changes as part of an ∼11-year cycle (see Figure 3.8; sunspot data collected from the World Data Center SILSO, Royal Observatory of Belgium, Brussels [37]). Solar cycles have been sequentially enumerated since 1755, with our dataset covering almost the entire duration of cycles 23 & 24 (1997 - 2019). Throughout each cycle, the sunspot number (SSN) roughly follows a Gaussian distribution, where the SSN provides a good measure of solar activity. The distribution's peak, and therefore activity, varies cycle by cycle. There is a close connection between the SSN and the number of type II bursts, even in the presence of a double peaking cycle such as cycle 24 [18]. Around 50 type II events were observed during 2002, cycle 23's most active year; and no events during 2009, its least active year. Relative to cycle 23, cycle 24 corresponded to a 38% decrease in type II events. It has been one of the weakest cycles since records began, making it a particularly poor period for producing data.

The varying levels of activity influences the occurrence of all solar events, which can make the challenges seen in Figures 3.5a & 3.5b more prevalent. If the levels of solar noise varies with time, then this could make it difficult for our detector to learn to ignore its corresponding features. Furthermore, because we only have access to cycles 23 & 24, it is difficult to validate that our detector will maintain its effectiveness in periods of stronger activity. It may also be the case that the appearance of type II bursts change enough between cycles to make their feature representations diverge from the original training context. Compared to cycle 23, shocks in cycle 24 survived over a larger distance from the sun; 60% of bursts in cycle 24 ended below 0.5 MHz, whereas only 42% did in cycle 23 [18].

In Chapter 5, we experiment with training separate classifiers for different levels of solar activity. The aim is to assess which performs better: reducing the variance caused by solar activity through activity-specific classifiers, or utilising all training samples in a single classifier to better generalise to the variances outside of solar activity.

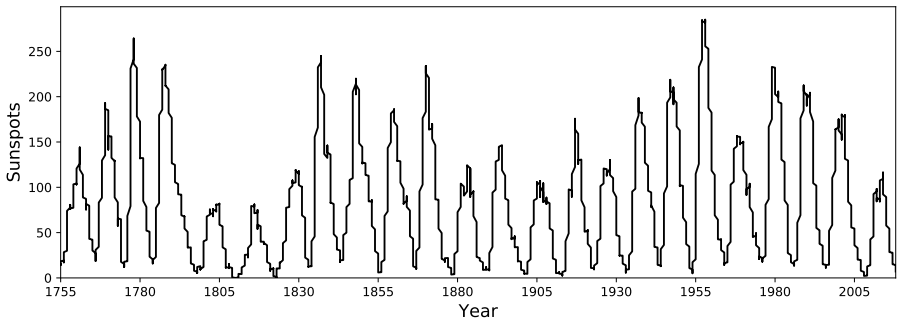


Figure 3.8: Solar cycles 1 – 24 (1755 – 2019).

4 Methodology

4.1 Overview

4.1.1 Data preprocessing

Preprocessing the data prior to feature extraction can help to alleviate the issues discussed in Section 3.2, though care is needed to ensure that the existing issues aren't being amplified even further. Noise removal techniques are likely to degrade the overall quality of the data, and in particular may have the greatest affect on the already problematic low contrast information. Similarly, contrast enhancement techniques are likely to worsen the influence of noise. Thus, our goal is to find the optimal trade-off between noise removal, preservation of signal, and contrast enhancement (see Section 5.2 for an analysis of this trade-off).

To remove the background noise, we propose to use a two-stage approach which specifically targets the Gaussian nature of the background. Our first stage consists of an intensity analysis to identify the values which fall within the intensity distribution of the background. We use a sigma parameter to partially remove the noise in favour of preserving the useful signal that overlaps within this intensity range. Our second stage aims to remove the remaining noise by removing small groups of spatially connected values. We choose not to target RFI or calibration signals for noise removal since they cannot easily be removed without being detrimental to the signal of interest. Contrary to the background noise, these sources of noise consist of predictable spatial patterns, so we instead rely on the classifier's ability to learn to ignore them. We attempt to restore missing measurements by applying a filter over the missing values to capture information from neighbouring samples.

The sensor data used in this study consists of continuous intensity values that are boundless, so it is necessary to normalise these values to a fixed range before we can make use of standard algorithms that expect unsigned bytes as input. A linear mapping cannot be used since extreme values would cause significant compression to the usable intensities. We instead use the sigmoid function to clip extreme intensities to the boundary values, which consequently allows the usable intensities to cover the full range of possible values. We also enhance the contrast by using histogram equalisation (HE) to spread out the intensity values. For general application, we give a rough guideline of how to choose suitable parameters, with a more in-depth analysis being given for Wind/WAVES in Section 5.2.

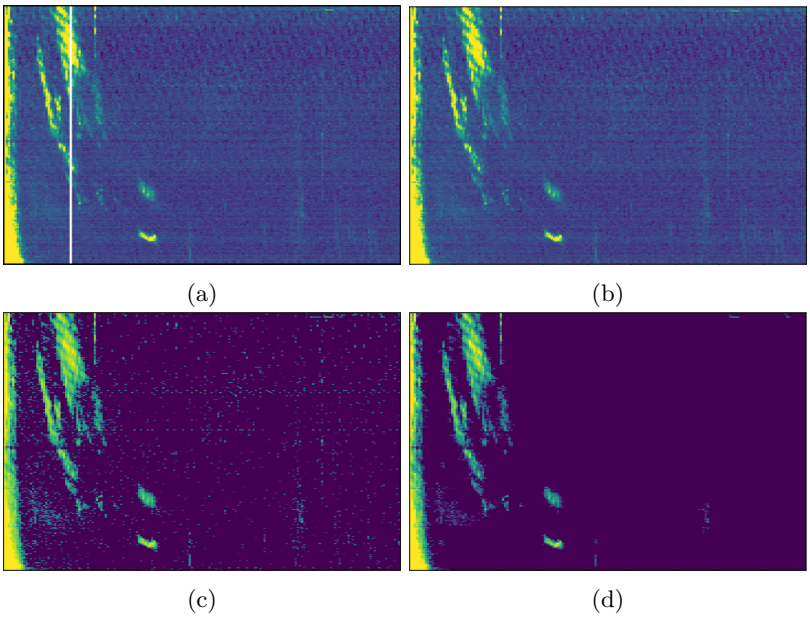


Figure 4.1: Preprocessing pipeline. a) Raw signal data with a period of no measurements. b) Applying a filter over the missing period to restore it. c) Removing a portion of the background distribution. d) Removal of small objects. For visualisation purposes, the intensities of a & b are clipped to 1.3, and the contrast of c & d has been corrected using our normalisation. In practice, we apply our normalisation after the background has been fully removed.

4.1.2 Physics-informed localisation

Given the geometrical nature of the bursts' structure in 2D space, we choose to approach their detection as an object recognition problem. However, we recognise that traditional object detection techniques such as as uniform sliding windows and pyramid scaling do not translate well to the variances seen in the bursts' spatial properties. This is because these properties are a consequence of their physical properties, with the most notable being how they drift through frequencies over time. We have seen in [2] how a burst's change in drift rate can be well described by a function of its frequency. Following [28], we choose to exploit this prior knowledge by integrating it into our detector so that the problem of shape recognition can be simplified; and thus, allowing the use of simpler machine learning models and fewer training samples. In their approach, they chose to transform the data with respect to the frequency dimension such that the variance of the frequency-dependency was removed. However, as they had only considered one of two spatial dimensions during transformation, they had lost information as a result of not being able to capture the full shape of the burst. Our method resolves this issue by considering both the frequency and time dimensions during transformation so that the full description of the bursts' curvature in time-frequency space is captured.

We propose to utilise the known drift model to constrain the searched region of interests (ROIs) to be in compliance with the curvature of type II bursts. We fully take advantage of this constraint by searching for curved ROIs that are directly focused on the bursts' signal. In addition, we transform the curved ROIs into a rectangular grid to create a more suitable data representation for feature extraction and classification. We demonstrate the effectiveness of integrating this prior knowledge by utilising a simple detection pipeline based on sliding Histogram of Oriented Gradients (HOG) windows and logistic regression. To overcome the discontinuities in the bursts' signal, we train our model to detect smaller segments rather than the full event at once. Figure 4.2 shows an example of the detection pipeline. Following detection, we segment the bursts by filtering out pixels with a low detection response. We also use the previous classification of the background to refine the segmentations. Figure 4.3 shows an example of segmentation.

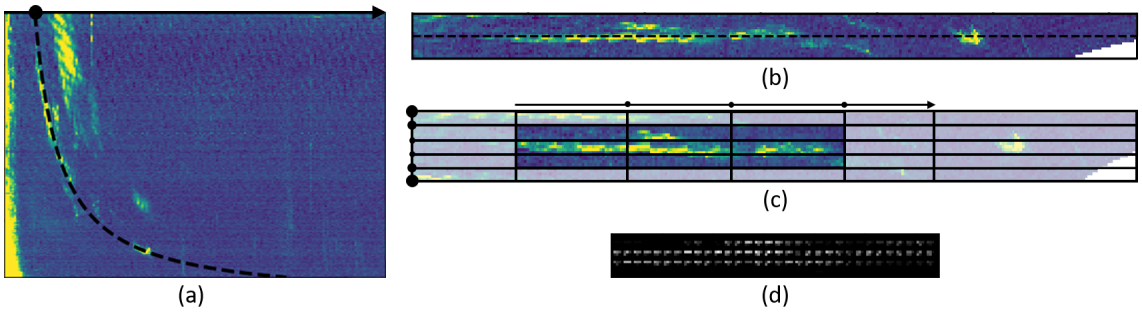


Figure 4.2: Detection pipeline. a) We use the drift model to scan 1D curves along the temporal axis that correspond to the curvature of type II bursts. b) We expand the curves outward into 2D space and transform the regions into rectangular grids. c) We use sliding windows of varying sizes to search for smaller segments in the grid. d) We compute and classify HOG features of the windows (example corresponds to highlighted window in 4.2c).

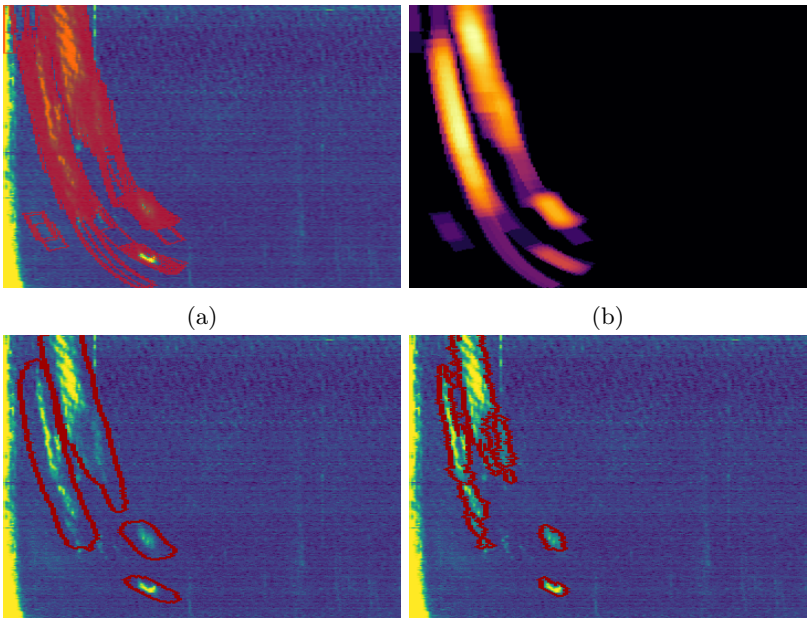


Figure 4.3: Segmentation pipeline. a) All detected windows in original image space. b) Density map of the windows. c) Discarding low density pixels to produce an initial segmentation. d) Refining the segmentation by discarding pixels that overlap with the background.

4.2 Data preprocessing

4.2.1 Background removal

Statistical subtraction

As in [35], we aim to estimate the background's parameters so that it can be subtracted away. The background distribution is Gaussian and varies depending on frequency, so in [35] they use the previous 7 hours to calculate the mean and standard deviation for each frequency. However, since extreme values may be several orders of magnitude higher than the background, we find the mean as a metric to significantly overestimate the parameters. Using the median as an alternative to the mean results in much better estimations, but only for channels unaffected by RFI. In order to estimate the parameters more consistently, we need to use an approach that is able to ignore non-background values. A simple solution would be to use all of our background windows to fit an idealised Gaussian distribution, and then constrain the metrics to only consider values within the resulting distribution. An even simpler solution would be to use the resulting distribution itself as the background subtraction parameters. Figure 4.4 shows this approach in effect using a single Gaussian for all frequency channels.

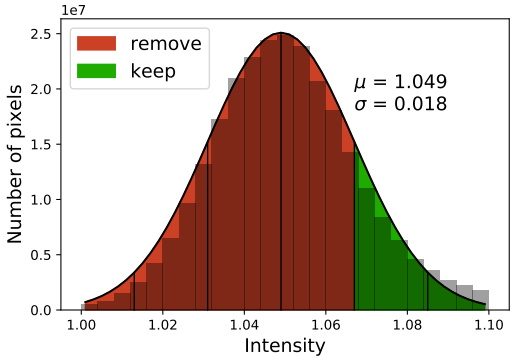
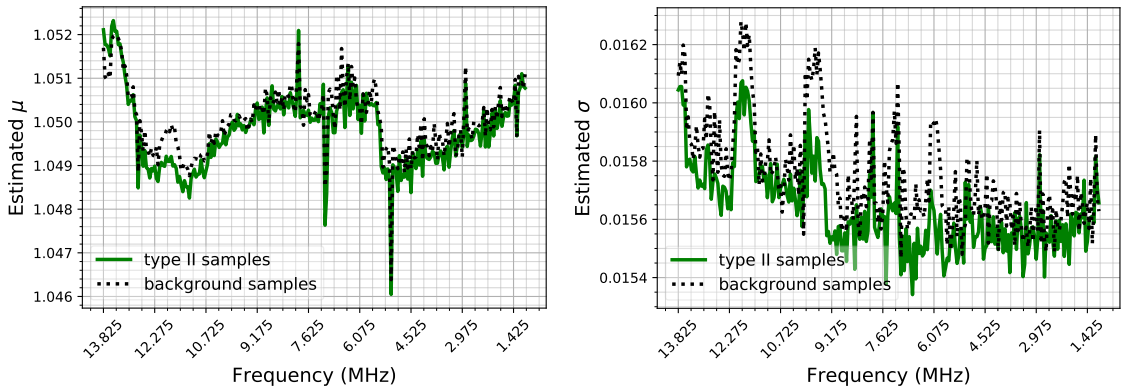


Figure 4.4: Distribution of background intensities (all frequencies). The global distribution can be described by a Gaussian curve ($\mu = 1.049$, $\sigma = 0.018$). A threshold (e.g. 1σ) is used to remove a percentage of the background's lower distribution.

However, any solution that relies on computing the parameters ahead of time are not robust against changes to the distribution over time, so a better solution would be one that makes no assumptions about the parameters of the distribution. Since we can make the assumption that the distribution will always be Gaussian, we can use this knowledge instead to constrain the parameters. Thus, we propose to use least squares to fit a Gaussian model to a histogram of intensities, where the resulting fit will be used to extract the parameters used for subtraction. After subtraction, any intensities below zero are clipped to zero. To compute the histograms, we allocate an empirical intensity width of close to half a standard deviation (0.008) to each histogram bin, using Equation 4.1 to calculate the number of bins to use. A histogram is computed for each frequency channel over the previous 12 hour period, where the choice of using 12 hours is to ensure enough information is available to estimate the parameters. If a fit cannot be made when attempting to fit the Gaussian, such as when a frequency is dominated by RFI, then we increase the resolution of the histogram bins to capture more data (we find a width of 0.001 to be sufficient). Using this approach, Figure 4.5 shows an aggregate of the resulting parameters on our background and type II samples.



(a) (b)

Figure 4.5: Estimated background parameters per frequency channel. It is evident that different frequencies have different background parameters, which the use of a global threshold would fail to capture. Because the peaks and troughs are independently captured in both classes, we can be confident that this is not down to error in the estimation approach. There is a reasonably large disparity in the estimated sigma for both classes, but since the pattern is preserved, it supports the belief that the parameters do change over time.

Spatial connectivity analysis

Increasing the sigma parameter of the background removal has diminishing returns to the amount of background noise removed, so we aim to avoid removing the full distribution of the background in order to preserve the overlapping type II signal in this range. After partially removing the background noise, the remaining background will appear as speckle-like noise, which can then be removed separately. We avoid using a filtering based procedure (e.g. median filtering) since the filter would be applied to both the background and the type II bursts, so we instead aim to target the background independently.

Most of the background at this stage should be represented by the constant 0, so we create a binary mask of our data to separate background from non-background by setting any non-zero values to 1. We then group the non-background pixels of the mask into structured objects, using the rule that any two pixels connected vertically or horizontally are considered to be part of the same object. A threshold is then used to discard all objects that do not meet a specified minimum size (Figure 4.6). Given the random nature of the noise, it is statistically improbable for the remaining values to form large clumps of connectivity. On the other hand, the spatially correlated signals of solar events should result in even isolated signals to make up much higher levels of connectivity.

0	5	5	5	5	0	0	0	0	0
0	0	0	5	0	0	2	2	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	6	6	0	4	4	0	0	0
0	0	6	6	0	4	4	0	0	0
0	0	6	6	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	3	3	0
0	0	0	0	0	0	0	0	3	0

(a) Original mask

0	5	5	5	5	0	0	0	0	0
0	0	0	5	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	6	6	0	0	0	0	0	0
0	0	6	6	0	0	0	0	0	0
0	0	6	6	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

(b) Processed mask

Figure 4.6: Removing objects with low connectivity. Background pixels are white. a) We group non-background pixels into objects and label them with their respective sizes. b) We use a threshold (e.g. 5) to discard objects that do not meet the threshold size.

4.2.2 Restoration of missing measurements

As previously seen in Figure 3.6, it is uncommon for consecutive periods to suffer from missing measurements. Standalone occurrences account for the majority of cases at 78%, with two or three consecutive samples accounting for 11% and 5%, respectively. Therefore, we can attempt to restore this data since we are very likely to have access to data in the immediate vicinity. For any instances of missing data, we place a small filter over the samples to capture information from neighbouring samples. For cases where we cannot successfully restore the data due to the consecutive period being too large, we set the samples to be the same constant value as the background. As a result, any periods of missing data should not cause any confusion to the classifier. We experiment with different filter types and sizes in Section 5.2.

4.2.3 Intensity normalisation

We use HE to enhance the global contrast of our data by spreading out its intensity values. Typically, HE is applied to images with n-bit integer channels, where the number of grey levels L is known to be fixed at 2^n. Since our data does not have a fixed representation, then we do not possess a parameter L to inherently enable the application of HE. However, we can derive the number of grey levels L_I needed for an input image I by considering each grey level to represent a fixed value width w

$$L_I = round(\frac{max(I) - min(I)}{w}). \tag{4.1}$$

To prevent extreme intensity values from causing L_I to be very large, we use the sigmoid function

$$\frac{1}{1 + e^{-gI}} \tag{4.2}$$

to map our data to $[0.5, 1]$ prior to HE, where g is a gain parameter used to control the strength of the intensity shift. In general, we find $w = \frac{g}{1000}$ to be a good value for enhancing weak signals, and the value of g should be chosen based on the expected intensity range of type II bursts. We can use the inverse of sigmoid as a rough guideline for assessing the suitability of the parameters by calculating the interval

$$[ln(\frac{0.5 + w}{0.5 - w})g,\ ln(\frac{1 - w}{w})g], \tag{4.3}$$

which represents the range of values that are able to satisfy the desired level of precision w in the binning process of grey levels. Assuming $g = 1$ and $w = 0.001$, then we get the interval $[0.004, 6.907]$, which is in agreement with our empirical observations of type II burst intensity values (after background subtraction). Another option for choosing w is to choose the lowest value that has reasonable demands on memory and computation time. The interpretation of reasonable is discretionary, and may be a factor of the real-time sampling rate, or the number of archived samples being evaluated. Figure 4.7 shows the performance when varying the number of bins. For our dataset, we find the optimal parameters in Section 5.2 by evaluating the parameters of intensity normalisation and background removal in unison.

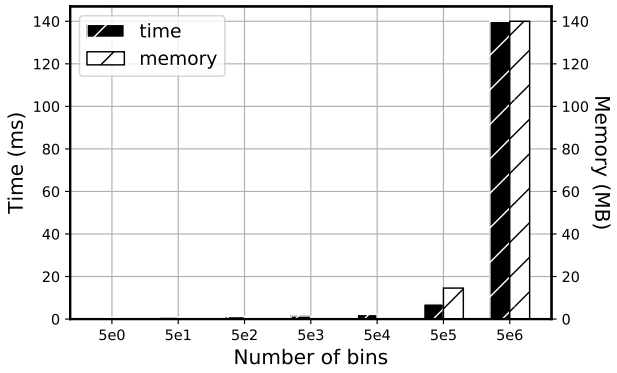
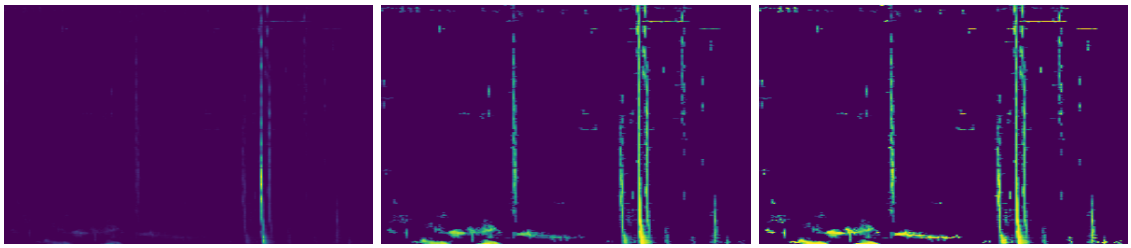


Figure 4.7: Growth of time and memory cost against number of bins used in HE. The performance remains steady for $L_I \leq 50{,}000$ and then begins to be much more relatively expensive for $L_I \geq 500{,}000$.

When scaling the intensity values from the continuous domain into the discrete domain, we test both global and temporal-wise scaling. Since the transformation of intensities during HE is globally monotonic, the enhancement of weak signals may be conditional on the overall distribution of intensities within the considered window. By applying the scaling to each temporal sample independently, we violate the monotonic rule in favour of increasing the intensity of weak signals. If we consider the fact that both the slope and intensity of type II bursts decrease with frequency, then the temporal samples that contain weak signals will typically contain less non-background signal. Therefore, the enhancement of these signals will be less restricted. An example of this can be seen in Figure 4.8. However, the presence of RFI (and possibly harmonics) may make this effect too inconsistent to be useful. In any case, HOG's local block normalisation will help to extract similar features from both low and high intensity bursts.



(a) No HE (b) Global scaling (c) Temporal-wise scaling

Figure 4.8: Effect of using HE with different approaches to scaling. a) Without HE, the intensity of the type III burst severely limits the dynamic range of the window. b) The use of HE allows the intensities to be more evenly distributed; the type II burst can now be seen. c) The effect of temporal-wise scaling is quite subtle, but the end result is a stronger contrast between the burst signal and the background.

4.3 Localisation

4.3.1 Physics-informed region of interest

Rectangular ROI

At test time, we constrain the ROIs to comply with the frequency-drift dependency of type II bursts. We use the drift rate model from Section 2.1.1 to define a relationship between frequency and offset in time, which enables us to derive the corresponding aspect ratio of our ROIs from the frequency range being evaluated. In doing so, we guarantee that any detections are constrained to the possible physics of type II bursts. However, within this set constraint, the traditional approach of using rectangular regions fails to take advantage of our prior physics knowledge to reduce the variance of the bursts' feature representations. Specifically, the signal at different frequencies presents variance with respect to its orientation, which consequently implies variance in the ROI's signal-to-noise ratio (SNR), where the point of maximum curvature contains the lowest proportion

of signal.

Curved ROI

To overcome these issues, we instead use the drift trajectory curve itself, i.e. the mapping between frequency and time offset, to model the shape of our ROIs. A 2D curved region is constructed from the 1D drift curve by expanding it in the direction of its normals. Due to the curved nature of the region, we ensure that the focus remains directly on the type II bursts themselves. Thus, we remove the variance of the SNR by maximising it at all frequency ranges. We also remove the variance of the bursts' shape caused by the frequency-drift dependency by straightening out the curved region into a rectangular grid. In all, this reduction of variance allows our problem to be greatly simplified. By normalising the bursts' shape — and therefore its resulting features — to be free from the influence of the frequency dimension, then the separation between type II and non–type II features should be much more apparent. Therefore, the use of simple machine learning models paired with our limited number of training samples should still be able to model an appropriate decision boundary. The transformed data representation also benefits from facilitating the application of standard image processing and computer vision techniques since these techniques are often inherently designed to work on matrices.

We straighten the curved region by parameterising the drift curve by arc length, which allows us to traverse the curve with respect to the distance travelled along it. We discretise the expansion of the curve by sampling both the curve and its normals at a rate of one sample per unit of distance. The transformation between the curved region to a rectangular grid then becomes a simple case of transposing the normals to columns in the grid. Because of the unit sampling, the dimensionality of the original and transformed ROIs are equivalent with respect to thickness and length. Therefore, we ensure that all information and spatial context is preserved. As a proof of concept, we use nearest-neighbour interpolation when sampling the pixel coordinates. An illustrated demonstration of the creation and transformation of the curved region is shown in Figure 4.9, and a precise definition of the process follows.

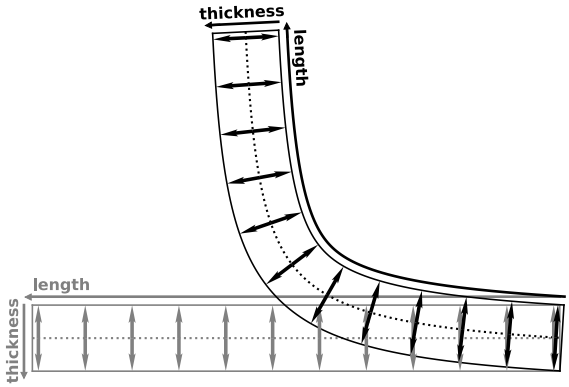


Figure 4.9: Creating and re-shaping the curved region. The drift trajectory curve (black dotted-line) is expanded along evenly spaced normal vectors in each direction to add thickness to the curve. The pixels are sampled along each successive normal vector to get the re-shaped output (grey).

Curved region: creation and transformation

Recall the drift rate model from Equation 2.1 in MHz/s, which states how the drift rate changes depending on frequency:

$$-df/dt = \alpha f^{\psi}, \tag{4.4}$$

where α and ψ is a scaling factor and power index on the frequency f, respectively. The *drift rate* describes the *rate* (in seconds) at which a burst *decreases in frequency* (in MHz). By inverting the model, we can use it to describe the cumulative increase in seconds from ∞ up to f

$$dt/-df = (\alpha f^{\psi})^{-1}. \tag{4.5}$$

Thus, for a range of frequencies, we can trace out the *drift trajectory* within time-frequency space. Let R_t (seconds) and R_f (MHz) represent the temporal and frequency resolutions of the data, as

well as I_t and I_f to represent the temporal and frequency indices of the data (image I). We can model the drift trajectory in image space by scaling α by R_t and substituting f as $f_{max} - R_f I_f$, where f_{max} is the maximum frequency of the data. Representing the drift trajectory as a mapping between I_f and I_t in image space is then given as

$$I_t(I_f) = (R_t \alpha (f_{max} - R_f I_f)^{\psi})^{-1}. \tag{4.6}$$

We can parameterise the mapping by introducing a parameter δ in place of I_f to get the following position vector

$$\vec{r}(\delta) = \begin{bmatrix} \delta \\ I_t(\delta) \end{bmatrix} = \begin{bmatrix} I_f \\ I_t \end{bmatrix}, \quad 0 \leq \delta < n, \tag{4.7}$$

which can be used to trace a curve of the drift trajectory in image space for all n frequency channels. We can then define a function which gives the distance travelled along this curve up to the value δ

$$s(\delta) = \int_0^{\delta} \|\vec{r}\,'(\delta)\| \, d\delta. \tag{4.8}$$

The next step is to find the inverse function $\delta(s)$ such that we can find δ from an arbitrary distance s. Let $\delta(s) \approx \delta\epsilon$, given that $\delta\epsilon$ satisfies the minimum value of $|s(\delta\epsilon) - s|$, $0 \leq \delta \leq \frac{n-1}{\epsilon}$, i.e. we integrate numerically by incrementing δ by a small constant ϵ to find the corresponding value of $s(\delta)$ that closest matches our input s. We then reparameterise the drift trajectory function $\vec{r}(\delta)$ by arc length such that we can find the pixel coordinates for distances travelled along the curve

$$\vec{r}(s) \approx (\vec{r} \circ \delta)(s), \quad 0 \leq s \in \mathbb{N} < \ell, \tag{4.9}$$

where ℓ is the total length of the curve defined as $round(s(n-1))$.

We compute the derivative of $\vec{r}(s)$ numerically to approximate the unit tangent vector, which we then rotate by 90° to get the unit normal vector

$$\hat{n}(s) \approx \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \vec{r}\,'(s). \tag{4.10}$$

Given a desired thickness τ of our ROI, we expand the curve in the direction of its normals by scaling $\hat{n}(s)$ by half of the thickness in each direction. We do this in τ steps such that we have one normal vector per unit of thickness, where each step is given by $0.5\tau - k - 0.5$, $0 \leq k < \tau$. Thus, we define a function $\vec{r}(k, s)$ which takes as input a two-dimensional step along the curve in terms of thickness and distance and outputs the corresponding pixel coordinates

$$\vec{r}(k, s) \approx \vec{r}(s) + (0.5\tau - k - 0.5)\hat{n}(s). \tag{4.11}$$

After rounding the coordinate values to the nearest integer, we map the original ROI to a $\tau \times \ell$ matrix T by sampling the pixels in our image I along the normals for each unit of distance along the curve

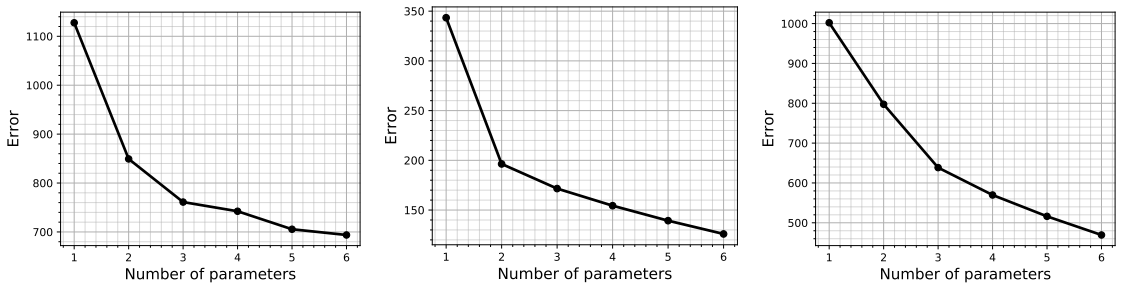
$$T_{ks} = I_{round(\vec{r}(k,s))}. \tag{4.12}$$

4.3.2 Parameter discretisation

In Section 4.3.1, we define a way of encapsulating the shape of type II bursts using three parameters related to their appearance: drift rate, length, and thickness. To utilise these parameters in a detection setting, we must map each of the parameter's continuous distributions into a fixed number of discrete parameters. Ideally, our discrete parameters should generalise well to the real distribution. We aim to achieve this goal by using optimisation techniques to minimise the error between our discrete and ground truth parameters. Our ground truth annotations do not state these parameters explicitly, so they must first be estimated from the pixel masks. We separate harmonics into their own annotations so that their parameters are evaluated independently.

Figure 4.10 shows how the error of our objective functions for each parameter decreases as we increase the number of parameters used. Using these trends, as well as some empirical reasoning based on perceived variance and the total number of parameter combinations we deem acceptable, we choose to use four drift rates, four lengths, and three thickness values. We increase our thickness values by 20% to account for imperfect fits of the drift rate, and then again by a factor of two to create a gradient between the burst and its surrounding background. These values are rounded to

the nearest integer divisible by two so that the ROIs can be conveniently cropped in a detection setting. Our final list of parameters are shown in Table 4.1. We note that after generating our training samples, we found only two events that corresponded to a fitting of a fifth drift rate. Therefore, we discard this value and keep the remaining four.



(a) Drift rate (b) Length (c) Thickness

Figure 4.10: Effect of number of parameters used on error.

Parameter	**1**	**2**	**3**	**4**
Scaling factor	1.47×10^{-4}	6.19×10^{-5}	9.54×10^{-5}	1.21×10^{-4}
Power index	0.25	0.58	0.82	0.5
Length (px)	42	74	132	164
Thickness (px)	5.33 (12)	7.42 (18)	10.28 (24)	-

Table 4.1: Discrete parameters chosen. Drift rate is composed of a scaling factor and power index pair. The initial thickness values are shown first followed by their adjusted values in brackets.

Drift rate

Given n desired drift rates, we define a $n \times p$ parameter matrix P which holds the parameters associated with each drift rate (its scaling factor α and power index ψ; $p = 2$). We initialise the state of each vector in P as $[5.5 \times 10^{-5}, 1.28]$, which is the reported best fit for DH type II bursts [2]. We use the Levenberg-Marquart (LM) algorithm [33] to optimise the state of P in two stages. First, we consider the definition in Equation 4.6 that defines the mapping from frequency to time in image space from the drift rate parameters. Let this be represented as $I(x, \vec{p}) = y$, where $\vec{p}$ is a parameter vector of the drift rate, x is the frequency index, and y is the offset in time in pixels for that frequency. To account for the arbitrary temporal position of the bursts within our set of annotations $\mathcal{A}$, we add an additional time offset term o to the mapping. For a given $a \in \mathcal{A}$ and parameter vector $\vec{p}$, the optimal o is found using LM by minimising the sum of squared errors

$$o(a, \vec{p}) = \underset{o}{\operatorname{argmin}}(\sum_{i \in a}(I(x_i, \vec{p}) + o - y_i)^2). \tag{4.13}$$

We then optimise the state of P using LM by minimising $\vec{e}$, which contains the errors for all annotations. The error for an annotation a is computed by evaluating the error for all drift rates in P and then taking the minimum error

$$e_a = \min_{1 \leq d \leq n} \{\frac{1}{|a|} \sum_{i \in a} |I(x_i, P_d) + o(a, P_d) - y_i|\}, \tag{4.14}$$

$$\vec{e} = \left[e_a^1 \; e_a^2 \; \ldots \; e_a^{|\mathcal{A}|}\right]. \tag{4.15}$$

Length

In Section 4.3.3, we create our positive training samples by sampling all possible segment combinations that meet our minimum SNR criteria. To define an objective function that complements our approach to sample selection, we evaluate our lengths within a similar context by cycling through the possible segment combinations and measuring the resulting SNR. For efficiency purposes, we

consider the 1D case of SNR, where we measure the proportion of noise (non–type II signal) within a window of the fitted discrete 1D drift curve.

Increased noise is a result of two possible outcomes: either the window's length is longer than the segment, or the window contains two segments with some noise in-between. We set the level of noise as an objective to minimise to ensure our lengths generalise to the creation of high quality samples. The error of samples containing too much noise is ignored to mimic the rejection of training samples that do not meet the specified criteria. Since the quantity of training samples is also important, we set a second objective to minimise the number of samples rejected. In addition, a third objective is used to minimise the level of truncation resulted from the selected lengths being too short to capture the full length of segments. The three objectives are weighted empirically based on certain desires such as emphasising quality over quantity. The exact procedure used for computing the error is detailed in Algorithm 4.1. We minimise the errors of all annotations simultaneously using differential evolution [38], and include the rejection criteria as a parameter to optimise.

Algorithm 4.1 Length error of annotation

Input
- *lengths* ▷ set of discrete lengths
- *allowed noise* ▷ maximum percentage of noise allowed within window
- *segment positions* ▷ starting and end positions of segments along the fitted curve

Output
- *error* ▷ combined error of noise, truncation, and sample rejection

Initialisation
- *list of noise*
- *list of truncations*
- *list of samples*

for all segments **do** ▷ starting segment
- *start* ← start position of starting segment
- **for** starting segment up to all segments thereafter **do** ▷ end segment
  - *end* ← end position of end segment
  - *real length* ← *end* − *start*
  - /* *discrete length* must be ≥ *real length* unless *max length* < *discrete length*, in which case we are forced to truncate the segment */
  - *discrete length* ← lowest length in *lengths* ≥ *min*(*real length*, *max length*)
  - **if** (*start*, *discrete length*) not in *list of samples* **then**
    - *list of samples* ← (*start*, *discrete length*)
    - *noise* ← percentage of noise within window [*start* : *start*+*discrete length*]
    - **if** *noise* ≤ *allowed noise* **then**
      - *list of noise* ← *noise*
    - **end if**
    - **if** *length* = *max length* **then**
      - *list of truncations* ← percent truncated from end segment
      - break
    - **end if**
  - **end if**
- **end for**

end for

$e_g \leftarrow 0$ **if** *list of noise* is empty **else** *mean*(*list of noise*) ▷ average noise %

$e_t \leftarrow 0$ **if** *list of truncations* is empty **else** *mean*(*list of truncations*) ▷ average truncation %

$e_r \leftarrow 1$ - (*length*(*list of noise*) / *length*(*list of samples*)) ▷ % of samples rejected

$error \leftarrow \sqrt{{e_g}^2 + (1.5e_t)^2 + (0.5e_r)^2}$

Thickness

We estimate the thickness of our annotations by taking the ratio between the number of pixels

before and after skeletonising

$$\widetilde{\tau} = \frac{|a|}{|sk(a)|}, \tag{4.16}$$

where $sk(a)$ is the skeletonised a. Given n desired thickness values, we initialise a vector $\vec{\tau}$ with the mean ratio of all annotations

$$\bar{\tau} = \frac{1}{|A|} \sum_{a \in \mathcal{A}} \widetilde{\tau}_a, \tag{4.17}$$

$$\vec{\tau} = \left[\bar{\tau}\ \bar{\tau}\ \ldots\ \bar{\tau}\right]. \tag{4.18}$$

Let $h(\tau)$ be defined such that the nearest element greater than or equal to τ in $\vec{\tau}$ is returned, or $\max(\vec{\tau})$ if no such element exists. We optimise the state of $\vec{\tau}$ by using Powell's conjugate direction method [34] to minimise the vector $\vec{e}$

$$e_a = |\widetilde{\tau} - h(\widetilde{\tau})|, \tag{4.19}$$

$$\vec{e} = \left[e_a^1\ e_a^2\ \ldots\ e_a^{|\mathcal{A}|}\right]. \tag{4.20}$$

4.3.3 Sampling training segments from annotations

In Section 4.3.2, we choose a fixed number of discrete parameters to use in our segment detector. Prior to detection, it is necessary to train a classifier with examples that match the same setting as the detector — however, our annotations only contain information with respect to pixel coordinates and not the parameters used in our ROIs. Therefore, in order to train a classifier with positive examples, we must first divide our annotations up into smaller segments that are constrained to our chosen discrete parameters.

To determine which of our discrete parameters fits an annotation the best, we use the same criteria used in Section 4.3.2 to measure the error of a fit. For drift rate and thickness, these measures correspond to Equations 4.14 & 4.19, respectively. The length parameter is different because there is a one-to-many relationship between an annotation and the number of segments it has. Given that both the length of segments and the length between segments is unconstrained, then it is impossible to consider each segment as an isolated instance when fitting our limited number of fixed parameters to them. For example, we may have two segments close enough together such that the second segment is automatically included in the fit of the first segment. Even in a scenario where each segment can be fitted perfectly with its real length, it is still reasonable to want to include nearby segments within the same fitting as this would provide the classifier with more information relevant to type II bursts. Figure 4.11 shows several valid contexts in which a set of segments could be fitted. Since there is no single correct solution, we consider all possible solutions when dividing our annotations up into training samples. This allows us to re-use segments to boost the number of samples we can train our classifier with, which in turn should help to increase our detector's robustness to the variance of different segment contexts.

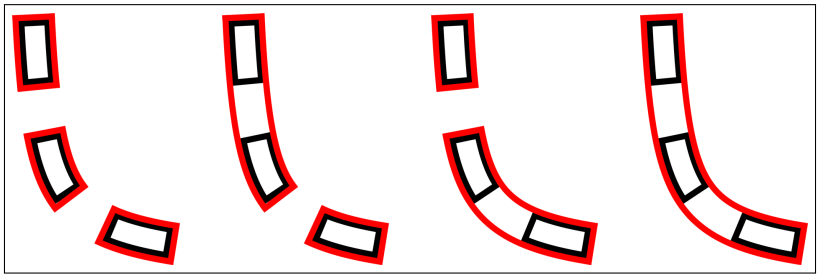


Figure 4.11: Fitting lengths to segments. Given the three segments shown, there are four possible variations in the context they can be fitted within. In total, there are six unique fittings.

After fitting a drift rate and thickness to an annotation, we cycle through all possible length fittings; we attempt to fit each of our lengths to each isolated segment in the annotation. Each attempt is either accepted or rejected through a criteria based on the Intersection Over Union (IOU) of the fitted ROI and the segment being evaluated (we use an empirical criteria of 30% IOU). This means that even for standalone segments, we may sample them many times versus only considering

the optimal sample for a particular segment. As long as the sample fits within our criteria, then it is possible for a similar sample to appear within our training set. Therefore, the additional information from sampling a segment within all possible contexts will be valuable to the classifier. To encourage the generation of samples that contain multiple segments, we relax the IOU criteria by also taking into account the level of intersection between the ROI and additional segments. Note that we only measure the intersection and not the union of other segments. If the union of the primary segment being evaluated is poor, then the resulting localisations will also be poor. However, if we only manage to capture a small portion of the second segment, then that can still be considered useful information to our classifier. For the implementation of selecting positive training samples, see Algorithm 4.2.

Algorithm 4.2 Selection of positive training samples for an annotation

Input
lengths ▷ set of discrete lengths
labels ▷ labelled mask of segments
segment positions ▷ starting positions of segments along the fitted curve
region coordinates ▷ coordinates of the drift trajectory region for a given harmonic
IOU threshold ▷ IOU threshold for accepting training samples
Output
list of samples
Initialisation
list of samples
for all segments **do**
start ← start position of *segment*
for all lengths **do**
/* Make sure the region is long enough from *start* to extract an ROI from ***/**
if *start* + *length* exceeds length of *region coordinates* **then**
continue
end if
ROI ← extract ROI from *labels* using *region coordinates* from [*start* : *start* + *length*]
IOU ← compute IOU between *ROI* and *segment*
/* If at least half of the IOU threshold is met, then we allow intersections with other segments to inflate the IOU score prior to the final check ***/**
if *IOU* ≥ *IOU threshold* ∗ 0.5 **then**
other intersections ← compute intersection % between *ROI* and all other segments
IOU ← *IOU* + *other intersections*
end if
if *IOU* ≥ *IOU threshold* **then**
list of samples ← (*start*, *length*)
end if
end for
end for

4.3.4 Detection

Traditionally, object detection techniques localise objects by sliding windows over the 2D Cartesian coordinate space (x and y axes) of the image [10]. In a similar fashion, we also slide windows over 2D space — however, we replace the y coordinate with one that directly relates to the shape of type II bursts: the arc length parameterisation of the burst's drift trajectory. As well as sliding our windows along a curved path, the windows themselves are also curved. We replace the width dimension with the curve's thickness in 2D space (by expanding the curve's normals), and height with the curved window's arc length. In summary, our ROIs can be described by four spatial properties similar to those found in traditional detectors, but have been adapted to relate to the shape of type II bursts. These adaptations are: temporal position (x position), position along the drift curve (y position), thickness (width), and length (height). The shape of an ROI in image space is dependent on how the last three properties construct the ROI from the 1D drift curve. Therefore, we also include drift rate, which controls the trajectory of the curve, as a fifth property

to be varied for our ROIs.

The implementation of our sliding window setting follows. Firstly, we scan our selected drift curves (that span the entire frequency range, shown in Figure 4.12) along the temporal axis. Using the maximum discrete thickness, each curve is constructed into a 2D curved region which is then straightened into a rectangular grid using the process in Figure 4.9. Since the transformed coordinate space of the rectangular grid is equivalent to the desired coordinate space of the curved region (with respect to length and thickness), all required operations — such as sliding windows and axis scaling — can be done on the rectangular grid with ease to map their effects onto the curved region. Axis scaling is required for creating the various length and thickness contexts of our ROIs, since we achieve this in the same manner as traditional detectors: by scaling down the space being searched (traditionally the image, but in our case the 2D drift region) so that larger objects occupy the space within the fixed-size sliding window. This process can be seen in Figures 4.13 & 4.14, which demonstrate the operations of scaling and sliding windows, respectively. The one-to-one mapping between the transformed region and the original curved region can also be seen. Since features are computed on the rectangular grid, the shape and resulting features of burst segments become invariant to their position along its drift trajectory.

We estimate the probability of each window containing a type II burst segment by classifying HOG features with logistic regression. The classifier is trained using the LIBLINEAR [13] library, with L2 regularisation and dual formulation. We also tried training a linear SVM but found performance to be near identical, albeit with significantly longer training times.

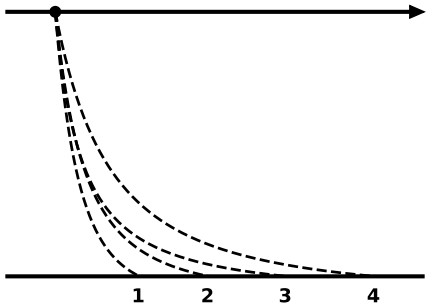


Figure 4.12: Temporal sliding. We slide all of our drift curves along the temporal axis in preparation for constructing the 2D region. Curve numbers correspond to the those in Table 4.1.

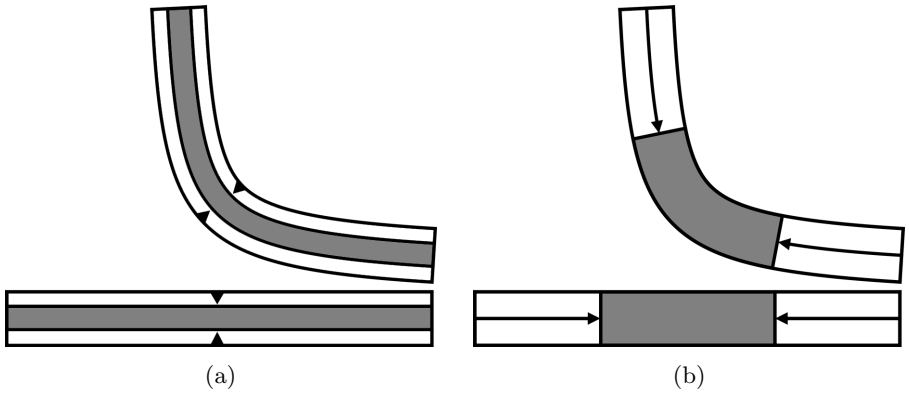
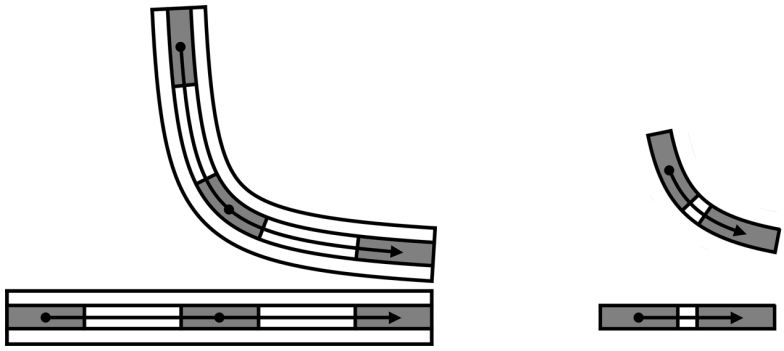


Figure 4.13: Axis scaling. White is the original region and grey is the down-scaled region. Down-scaling the curved axes can be thought of as a compression along the direction of the curve. a) Thickness axis. b) Length axis.



(a) (b)

Figure 4.14: Sliding windows along the drift region. a) Sliding window on the original scale; b) same sliding window on the down-scaled region.

4.3.5 Post-processing and segmentation

We begin by filtering out the input ROIs estimated with a low confidence score. In Section 4.3.2 we padded our ROIs with some additional thickness, and in Section 4.3.3 we allowed some extra length to be added onto our segments used for training. At this stage, we reverse both procedures by cropping the thickness and truncating the length of all ROIs. The amount of extra length appended is variable, so we test the optimal amount of truncation in Section 5.3. After this, we reconstruct the remaining ROIs back into image space (e.g. Figure 4.3a). Since object detectors are prone to detecting many ROIs for a single object, we must process all candidates with the aim of keeping only the most robust detections. A common approach is to process sets of overlapping ROIs, where either the non-maximum ROIs are suppressed, or an aggregate of the ROIs is computed from their parameters. However, since our classifier is capable of detecting multiple segments at once, then the corresponding gaps between segments will be present in the ROIs (e.g. Figure 4.11). Therefore, neither of these approaches are well suited for isolating individual segments.

A solution to this problem is to compute an aggregate of the ROIs independently of their parameters. We choose to do this using density-based aggregation. Specifically, we tally up the number of times each pixel has been detected (i.e. pixel voting, see Figure 4.3b for example) and suppress the pixels with a low total using an empirical threshold (e.g. Figure 4.3c). If we consider the initial problem of needing to post-process candidate ROIs, the problem itself stems from the degree of uncertainty associated with localising objects precisely. More optimistically, we can say that it stems from a classifier's ability to make a reasonable assumption despite the imprecision. There are two possibilities of imprecision: either the detection is off-centre, or its corresponding parameters are poorly matched. In practice, this is beneficial for generalising to the continuous nature of the real world through the restricted finite scope of the detector. By considering the density of all candidate ROIs, we effectively average out all aspects of imprecision by refining all detected pixels towards the true continuous representation. Therefore, we are able to aggregate ROIs of fixed parameters into flexible pixel level segmentations.

In general, we can assume that the refinement of pixels will converge to a single point as the voting threshold increases. As we lower this threshold to produce higher quality segmentations, one caveat is that we may introduce some weakly voted false positives. To overcome this, we use a two-stage process which first detects segments using a high threshold, followed by segmentation using a lower threshold. Any segments not detected in the initial detection stage are discarded. We also discard pixels that overlap with the background known from Section 4.2.1. This can help to improve the generalisability of the chosen thresholds by correcting segmentations that spill over the edge (e.g. Figure 4.3d). We test the optimal configurations regarding detection, segmentation, and background removal in Section 5.3.

5 Experiments

5.1 Training set

In preparation for our experiments on activity-based classifiers, we group our events based on the sun's level of activity during the time of the events. We use the recorded SSNs to do this, which we collect from the World Data Center SILSO, Royal Observatory of Belgium, Brussels [37]. Specifically, we use data that contains one sample per month, where each month has been smoothed over a 13-month window. This smoothing is standard procedure for scientific analyses performed on solar cycles [37]. Using the recorded SSNs over the duration of our data, we fit a Gaussian curve to each cycle so that they can be split into three groups of solar activity. The splitting points are defined by assigning each activity group to a fixed distribution within the Gaussian curves; we use the central 40% for high, the outer 20% for low, and the remaining 40% for medium. This process can be seen in Figure 5.1. The Gaussian splitting points have been verified by an expert as being representative of their respective solar activity groups, and the legitimacy of using this approach as opposed to other approaches that use the SSN directly has also been verified. This approach also provides a more favourable setting for our tests, since the relative distribution of future solar cycles can always be accounted for, but the absolute values of SSN (i.e. a cycle's level of activity) cannot.

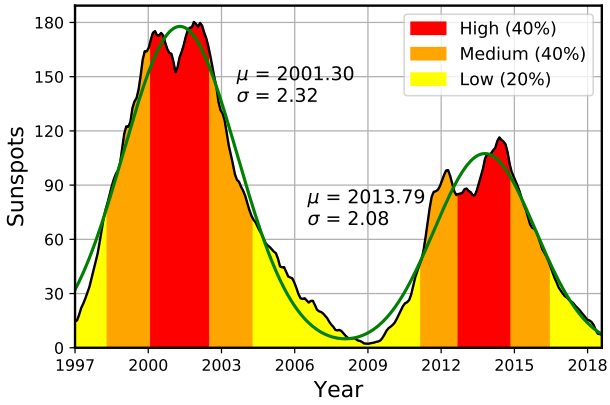


Figure 5.1: Splitting of solar cycles into high (red), medium (orange), and low (yellow) activity periods based on Gaussian fitting.

Our positive training samples are generated from our ground-truth pixel annotations using the method detailed in Section 4.3.3. For our negative samples, we use stratified random sampling to preserve the distribution of properties found in our positive samples. During sampling, we treat our negative sub-classes (type III bursts and background) distinctly, which in effect creates a 2:1 ratio between the negative and positive distributions. Firstly, we consider the number of event windows and the distribution of solar activity levels within those windows. We sample a (stratified) random selection of negative events from our event catalogues created in Section 3.1. The final distribution of event windows for each class can be seen in Table 5.1. Within these windows, we randomly select training samples that are stratified towards the parameters (drift rate, length, and thickness) within our positive samples. We let the starting position of our samples along the drift curve to vary freely, and we also let the temporal position to vary freely for our background windows. For the type III windows, we constrain the temporal position of our samples to be centred around the type III burst event. To add some variation to our training samples, we relax the constraint by allowing the temporal position to vary by up to 40% of the sample's thickness (40% is chosen fairly arbitrarily based on the perceived thickness of type III bursts). Four samples are randomly selected from each negative window to roughly match the total number of positive samples. Because we aim to remove the background noise in Section 4.2.1, we check our random samples against our strongest background removal to make sure some data is still retained. If not, then we re-sample until our criteria is satisfied. An additional set of negative samples is created through a single iteration of hard negative mining; we increase the size of our negative set by 25% by choosing the highest probability detection from each negative window (such that we now have 5 samples per negative window as opposed to the original 4). This approach ensures that the same

distribution of solar activity is preserved. The final distribution of samples and their parameters is given in Table 5.2.

	All	Low	Med	High
Type II	**244**	49	89	106
Type III	**242**	47	89	106
Background	**245**	48	88	109

Table 5.1: Class distribution of event windows. A breakdown of the number of windows per activity group is given.

	All	Drift				Length				Thickness		
		1	2	3	4	1	2	3	4	1	2	3
Type II	**982**	19%	47%	15%	19%	36%	31%	18%	15%	37%	37%	26%
Type III	**1210**	20%	42%	18%	20%	47%	27%	14%	12%	33%	35%	32%
Background	**1225**	17%	41%	22%	19%	43%	26%	16%	15%	31%	34%	35%

Table 5.2: Class distribution of training samples. For each parameter, the table shows a breakdown of how much their discrete values (shown in Table 4.1) contribute to the number of class samples. It is evident that the hard negative mining has brought an imbalance to some of the parameters; the lowest length and highest thickness is more likely to produce false positives.

5.2 Preprocessing and feature extraction

5.2.1 Evaluation and parameter tuning criteria

In total, we test the effects of 16 parameters spread across the following four procedures: background removal, intensity normalisation, missing measurement filtering, and feature extraction. We tune the parameters using a hierarchy of three grid searches: background removal and intensity normalisation first, filtering second, and feature extraction last. The procedures of background removal and intensity normalisation are merged due to the strong interdependence of their effects.

To evaluate the performance of our parameters, we perform image classification with 10-fold cross-validation on our training samples. Samples are grouped by their respective event windows so that they always appear within the same fold. This ensures there is no contamination between the training and test sets. The F-score metric (Equation 5.3) is used as the evaluation criteria due to its summary of precision and recall (Equations 5.1 & 5.2), both of which we aim to maximise. Because the number of training samples within the activity-based subsets are quite low (as low as 20% of the original dataset), we choose to fine-tune a single set of parameters as opposed to one parameter set per classifier to avoid the risk of overfitting. We consider two scores: one for a general classifier trained on the entire dataset, and another for a specialised classifier composited of three distinct classifiers trained on solar activity–based subsets. The mean of both scores is used as the final evaluation criteria. On top of the existing cross-validation used, this approach helps to provide an extra layer of confidence that the resulting parameters are robust to different training and testing conditions.

$$Precision = \frac{TP}{TP + FP}, \tag{5.1}$$

$$Recall = \frac{TP}{TP + FN}, \tag{5.2}$$

$$Fscore = \frac{2}{Precision^{-1} + Recall^{-1}}, \tag{5.3}$$

where TP is the number of true positives, FP is the number of false positives, and FN is the number of false negatives.

We perform an initial round of fine-tuning on our dataset prior to hard negative mining, in which the resulting parameter set is used to search for hard negatives. After adding the hard negatives

to our dataset, a second and final iteration of fine-tuning is performed. The final parameter set can be found in Table 5.3. Between iterations, the optimal HOG parameters remained the same. Thus, the tests performed in Section 5.2.2 use the HOG parameters shown in the table, and the tests performed in Section 5.2.3 use the preprocessing parameters shown in the table.

Feature extraction	**HOG**
Cell size (px)	4^2 (4×4)
Block size (cells)	3^2 (12×12)
Block overlap	$^3/_4$ (3×3 stride)
Bins	10
Gamma correction	No
Signed gradients	Yes
Length padding (px)	12
Window sigma (px)	8
Normalisation	L2-Hys (0.2 clip)

Background removal	**Adaptive**
Sigma	0.5
Min size (px)	5
Intensity normalisation	**Sigmoid+HE**
Gain (g)	10
Bin width	$^g/_{1000}$
Filtering	**Mean**
Height (px)	5
Width (px)	3

Table 5.3: Optimal parameter configuration. Left) Feature extraction parameters. Right) Preprocessing parameters.

5.2.2 Preprocessing

Normalisation	Adaptive	Fixed	None
Linear	82.63	82.92	82.95
Sigmoid	91.15	90.72	88.77
Sigmoid+HE	**91.47**	91.13	90.34
Sigmoid+HE (TS)	91.32	91.06	90.68

Table 5.4: Performance of various background removal and intensity normalisation procedures. TS=temporal scaling.

Background removal
Table 5.4 shows that adaptive background removal outperforms fixed removal in all non-linear approaches, with the advantage being in the range of 0.2 – 0.5 F-score. In cases where the adaptive approach is inconvenient to use, such as for ground-based instruments that can only observe the Sun for half the day, then the use of a fixed approach still remains to be a viable alternative. For HE (the best performing normalisation procedure), the effect of altering the background removal parameters is shown in Figure 5.2. Results for other normalisation procedures can be found in Appendix 2. Both adaptive and fixed removal have similar trends throughout, where a low–medium sigma threshold paired with a medium object size threshold generally provides the best performance.

Understandably, the sigma threshold is the axis of greatest variance due to its aggressive approach of discarding a select range of intensity values entirely. The performance generally receives a substantial drop after 1.5σ. For HE, Figure 5.3 shows the optimal trade-off between a too conservative threshold versus a too aggressive one. The threshold's effect on recall is the same for both adaptive and fixed removal, where the performance gradually climbs to a peak followed by a steady decline. The result of targeting the background too aggressively clearly has a significant impact on the ability to detect weaker burst segments. The peak recall for both approaches occurs at an offset, where the optimal thresholds are 0.5σ and 1σ for fixed and adaptive, respectively. This is likely due to the higher estimate of the standard deviation parameter for fixed removal compared to adaptive (Figures 4.4 & 4.5b). The same offset can also be seen in the false positive rate (FPR), where a surprising spike occurs at the threshold of optimal recall. This effect is three-fold for the adaptive approach, with a substantial ∼9% increase occurring relative to its neighbouring thresholds, compared to a ∼3% increase for fixed. Consequently, this results in an adaptive 0.5σ to have the highest score despite its lower recall. The cause of the spike in FPR is not apparent, but a possible explanation is that enough weak type II signal is being removed such that its features are similar to the remaining background. In this scenario, a slightly lower or higher threshold would

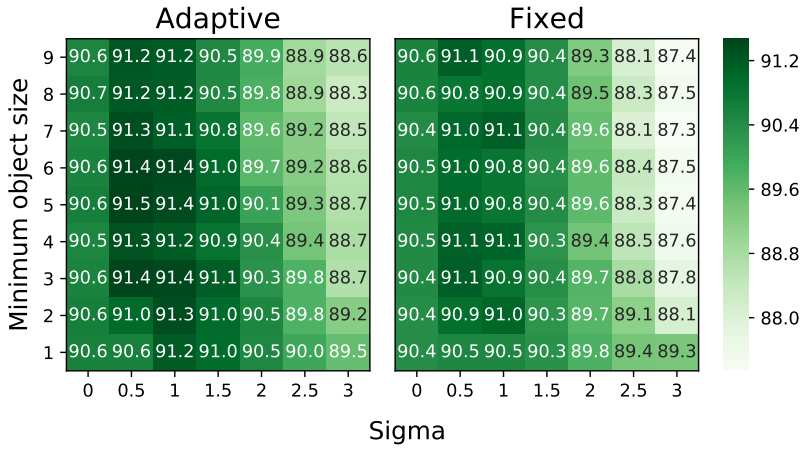


Figure 5.2: Effect of background removal parameters (Adaptive vs Fixed Gaussian estimation). Scores are aggregated by selecting the best performing HE parameters for a given combination of background removal parameters.

either preserve enough type II signal, or remove enough background, such that their corresponding features are more separable. A high threshold also appears to be associated with an increased FPR.

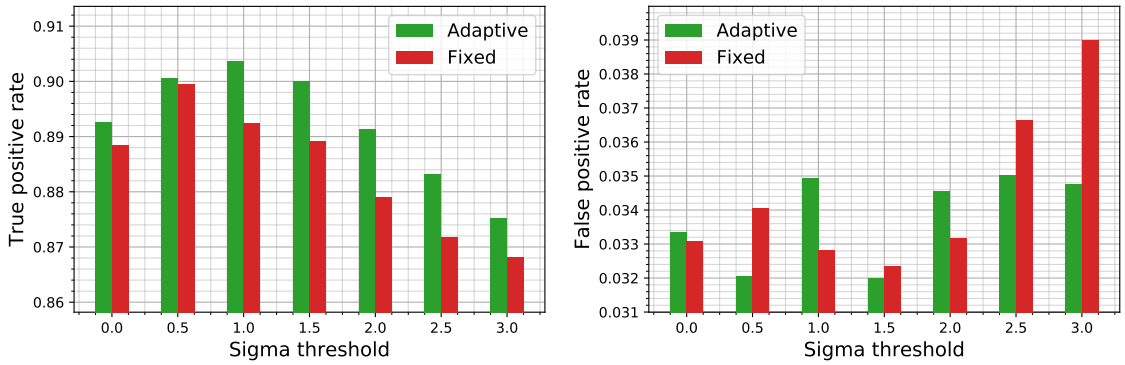


Figure 5.3: Effect of sigma threshold for adaptive and fixed background removal. Left) effect on true positive rate. Right) effect on false positive rate.

Relative to the sigma threshold, there is less variance when altering the minimum object size threshold because the effect is contextual rather than absolute. Assuming that the intensity of burst signals are above the sigma threshold, then their spatial connectivity will likely be high enough to be preserved. We can see that as the sigma threshold increases, there is a compounding effect wherein the spatial connectivity of burst signals breaks down enough to be removed in the second stage. Thus, in general, the variance of the object size threshold is positively correlated with the sigma threshold. However, for the optimal threshold specifically (adaptive 0.5σ), the range in scores is as high as the range of a 3σ threshold. Whereas the variance along 3σ highlights the importance of preserving the burst signal, the variance along 0.5σ highlights the importance of removing the background. This is reinforced by the observation that between an object size threshold of $1 - 3$, we see a difference in F-score of 0.8 for both 0.5σ and 3σ. However, the direction of change is inversely correlated, indicating the difference between background removal and preservation of burst signal.

Intensity normalisation

Table 5.4 shows that a naive linear approach to normalisation directly results in a substantial decrease in classification performance: between 6.6% – 9.7%. The resulting dynamic range of the image is likely to be much higher than the intensity range of type II bursts — so much so that a large portion of type II burst signal may be removed entirely. The use of temporal scaling during

HE reduces performance compared to global scaling, with the exception of when no background removal is used. Performance is consistently improved by using HE on top of sigmoid normalisation, especially when no background removal is used. Figure 5.4 shows the true strength of HE, where the use of sigmoid normalisation alone makes achieving good performance dependent on choosing the correct gain parameter. However, by using HE to balance the distribution of intensity values, we effectively eliminate any variance associated with the gain parameter. When looking at the effect that bin width has on performance, we begin to see a significant reduction when using a precision level that's too low. However, when using a reasonable level of precision, any variance between the parameters becomes relatively minor. Using a bin width of at least $g/1000$ results in the maximum difference in score to be 0.12. Thus, the normalisation procedure should be much more easily adaptable to other instruments since good performance is not dependent on the parameters chosen.

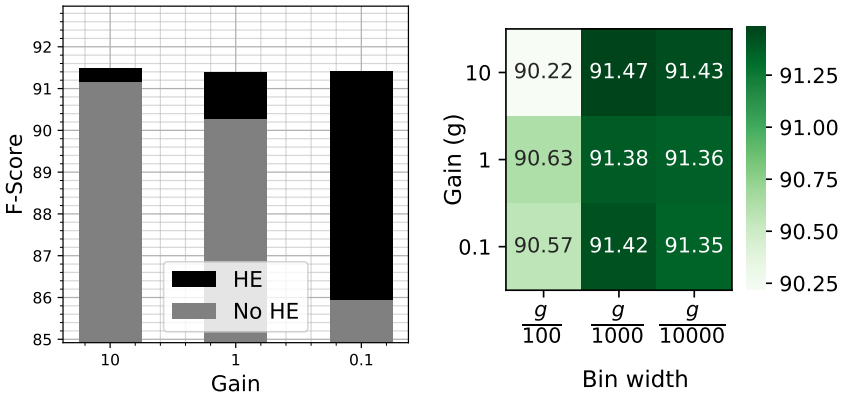


Figure 5.4: Effect of HE parameters. Left) Comparison of sigmoid normalisation with and without HE. Right) Influence of histogram bin width.

Filtering

We test three types of filtering procedures to capture neighbouring information as a replacement for missing measurements: median, mean, and max. In general, max filtering tends to degrade performance, median filtering preserves performance, and mean filtering boosts performance. The degree of change is relatively minor, with a range of -0.22 – 0.21 F-score. In all cases, a width of 3 performs optimally, but the optimal height varies depending on the chosen width and filtering approach.

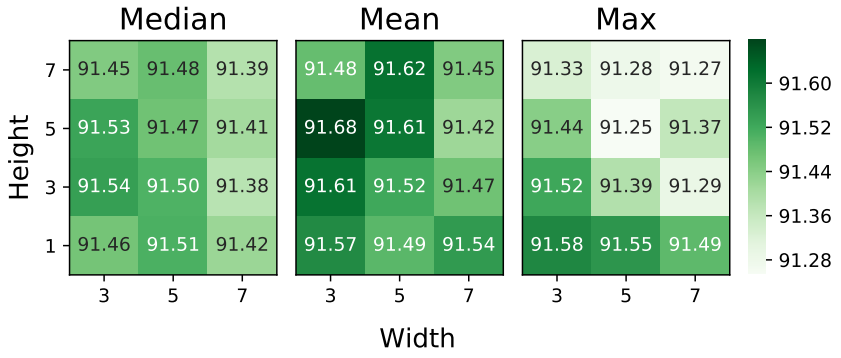


Figure 5.5: Effect of filtering parameters.

5.2.3 Feature extraction

As in [10], we use Gaussian spatial window smoothing with a sigma of 8 pixels, and L2-Hys block normalisation with a clipping value of 0.2. They find square blocks to be more effective than rectangular ones, with blocks of size 2^2 or 3^2 to be the best. Larger blocks are less adaptive to local imaging conditions, and small blocks (1^2) are unable to capture valuable spatial information.

With this in mind, we restrict our tests to use blocks of size 2^2 or 3^2. Cells of size 6^2 are found to work best for human detection, which they expect to be application dependent. Our ROIs are much smaller at a height of 12 pixels, so we test cells of size $2^2 - 4^2$ to ensure compatibility with our block sizes. They also find overlapping blocks during normalisation to be an important factor for performance, so we test an overlap of $1/2 - 3/4$. The optimal number of histogram bins was found to be 9, so we test 9 ± 3 bins. We also test the use of gamma correction and signed gradients, as well as padding our ROIs length-wise with $6 - 18$ additional pixels. The effects of varying all parameters can be seen in Figure 5.6.

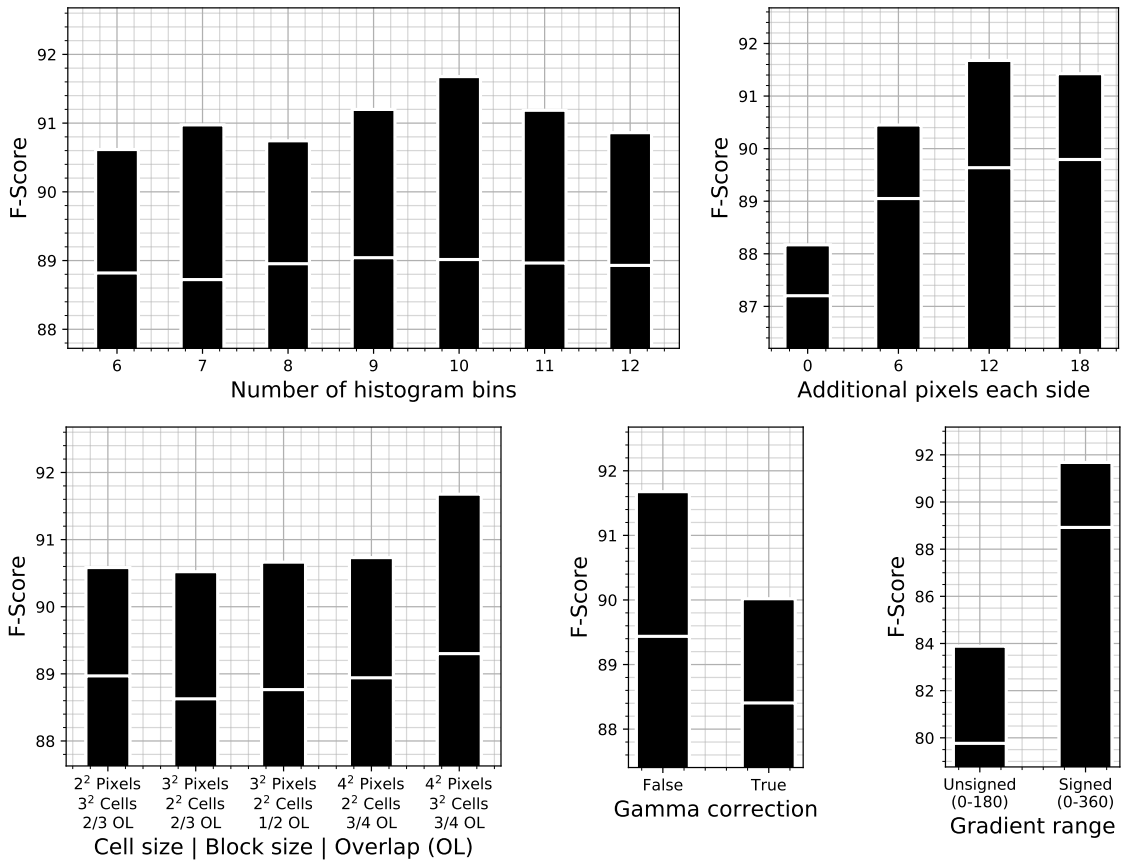
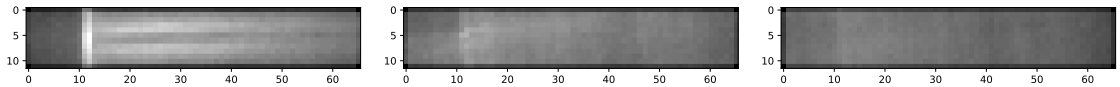


Figure 5.6: Effect of HOG parameters. Each bar represents the highest score achieved for a given parameter, and the white marker line shows the mean performance across all parameter combinations. When showing the mean score, we fix the gradient range to use signed gradients, with the exception of the gradient range plot itself.

Out of all HOG parameters, the parameter with the single biggest impact is by far the use of signed gradients. By utilising the additional information in the $180 - 360$ gradient range, performance improves by $\sim$12% on average, or $\sim$9% when the optimal configuration is used. Because HOG cues mainly on shape contours, we can expect the descriptive power to be highest around the transition between burst signal and background. The thickness of our samples have been explicitly padded to capture this transition, and the effect this has on our type II samples can be clearly seen in Figure 5.7a. Two distinct horizontal strips can be seen where both sides of the burst transitions to the background. The first strip should represent positive gradients, and the second strip negative gradients. Thus, the inclusion of signed gradients means that the classifier not only expects a response around the two strips, but also a match between a given strip and its respective sign. This can help to reduce false positives against strong responses to off-centred signals such as type III bursts, since our transformed coordinate space will typically distort their shape to curve upwards. Since the type III burst would be at the edge of the window rather than the centre, any gradients around the top strip would have the incorrect direction.



(a) Type II (b) Type III (c) Background

Figure 5.7: Average gradient response of training samples by class. Training samples have a length of 66 and a thickness of 12, and these dimensions will remain the same throughout all future examples (Figures 5.11, 5.13 & 5.14)

As stated above, it is important that HOG is able to capture the transition between burst and background. In Figure 5.7a, we can also see a strong response where the training samples have been padded length-wise. However, there is also a less intense gradient that spans the entire thickness and can be seen in all class samples. This is because we are forced to add dummy values — which we choose as 0 to match the background — in place of non-existent coordinates when expanding ROIs at the boundary. The resulting vertical strip is an undesirable effect, and the gradient response of the strip is ∼19% greater in boundary type III samples compared to the average type II sample. These type III samples account for ∼5% of all negative samples, so the aim is for the classifier to learn when to ignore these gradients in the context of supplementary information. A secondary benefit of length-wise padding is that bursts longer than the length of its window are able to supply the classifier with more information. We can see this effect in Figure 5.7a, where the horizontal strips extend to the end of the padded window. This likely explains why a padding of 12 pixels performs better than 6 pixels, although performance starts to decrease as padding is increased further, indicating that on average the additional pixels become less relevant. Interestingly, the use of further padding after 12 pixels does actually perform better on average, just not when the optimal configuration is considered. When using the optimal configuration, performance increases by ∼4% relative to the use of no padding.

The use of gamma correction decreases performance by ∼2%. Since we already apply an intensity transformation prior to feature extraction, using gamma correction on top of this likely serves no meaningful purpose. Furthermore, the original inclusion of gamma correction was likely tailored to combat the variances seen in natural images, such as illumination and shadows, which are not present in our data.

The performance of the different block configurations are relatively steady except for the optimal configuration, which performs ∼1% better than the next best configuration. The only difference between the two configurations is the use of a 3^2 block size instead of 2^2, where a 4^2 cell size and $3/4$ overlap is used in both cases. The result is a 125% increase in block size, which in fact allows the block to span the entire 12 pixel height of the ROI. The ability to utilise all of the information available along the thickness axis may be the reason for the fairly significant performance increase.

The number of histogram bins is the parameter with the least amount of variance, so choosing the optimal value is not as crucial. When using a random configuration, almost all choices perform equally well. However, the optimal value does still perform ∼0.5% better than the second best, and the total margin is ∼1.2%. The optimal range is within $9 - 11$ bins, with the middle value of 10 being the peak. Using 10 bins, Figure 5.8 shows the overall contribution of each bin by class. As expected, the bins that contribute the most votes to the type II class are around the 90° and 270° mark (i.e. the vertical gradients). This is also true for the type III class — however, there is a reasonably large disparity between type II and type III samples within the bins centred around 90° and 270°. In fact, the property of having the vertical gradients perfectly centred within their respective bins may be the reason why 10 bins provides optimal performance.

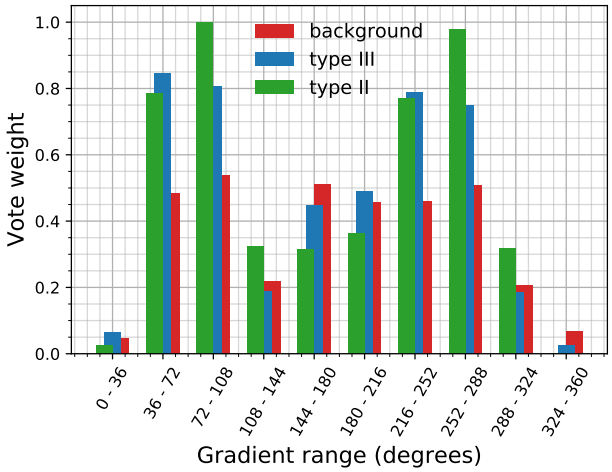


Figure 5.8: Vote weightings of orientation bins by class. Note that the chart shows votes accumulated using binary edge voting, as opposed to the interpolated voting used in the actual implementation of HOG.

5.2.4 Solar activity–based classifiers

Table 5.5 shows a breakdown of results for solar activity. We can see that the use of a single general classifier outperforms a composite of specialised classifiers by 1.3 points. However, we do see quite a wide variation in scores for the specialised classifiers: low activity has 1.11 points ahead of the general classifier, and high activity has 3.35 points below — equating to a total range of 4.46 F-score.

Classifier	Precision	Recall	F-Score
General	93.47	91.24	92.34
Specialised	92.25	89.86	91.04
Mean	92.86	90.55	91.68
Low	92.86	94.05	93.45
Medium	93.90	90.28	92.06
High	90.56	87.47	88.99

Table 5.5: Performance of activity-based classifiers.

Performance is seen to decrease with activity, although a secondary factor to consider is that a decrease in activity also corresponds to a decrease in training samples. Consequently, it is possible that the variation in performance may not be reflective of the variation in solar activity, but may instead be a result of the uncertainty associated with evaluating a small number of samples. We therefore evaluate whether the variation in performance is statistically significant by measuring the uncertainty of the general classifier at lower sample sizes. However, even if solar activity is indeed responsible for the variation in performance, the same variation may also be present, regardless of training conditions. Therefore, we also evaluate the general classifier's performance on each activity period separately.

Figure 5.9 showcases the two aforementioned evaluations. We can see that the variation in performance is in fact associated with solar activity, and that the effect is also mirrored when training a general classifier. As a result, we can conclude that the performance decrease of a specialised classifier is solely due to the decreased number of training samples, and that no combination of mixed classifiers would be beneficial. With that said, the performance increase relative to the general dataset is higher when using specialised classifiers for both low and medium activity. Performance on the low activity dataset is 2.28% ahead of the general dataset when using a general classifier, but the use of a specialised classifier results in performance to be ahead by 3.86%. Performance of the general classifier appears to plateau given enough training samples, so if the increased relative performance for certain specialised classifiers allow performance to scale better with more data,

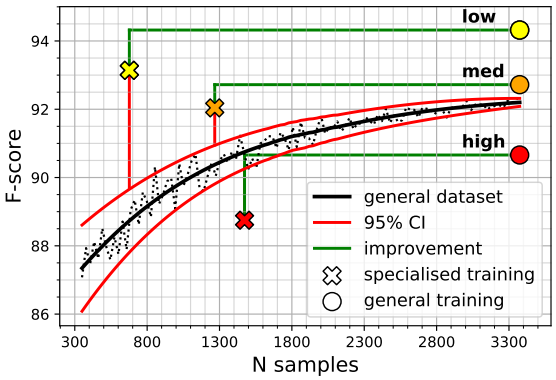


Figure 5.9: Activity-based classification performance. We compare the performance of specialised classifiers on activity-based datasets (cross) against the performance of a general classifier on activity-based datasets (circle). For reference, we plot the performance of a general classifier on the general dataset (black) for varying amounts of training samples. Data points (dotted) represent 10 trials of random sampling with a class ratio of 4:5:5 (type II, type III, background); 10-fold cross-validation is used for each trial. A 95% confidence interval (red) is shown for a random sample to fall within the performance distribution of the general classifier and dataset.

then it is possible that some degree of activity specialisation may eventually outperform a general classifier.

5.2.5 Failure-case analysis

ROI parameters

Figure 5.10 shows classification accuracy broken down by class and parameter value. The average appearance of training samples by class and parameter can be seen in Appendix 3 for reference. Type II samples perform the worst with an accuracy of 91%, compared to 96.9% for type III samples and 97.9% for background samples.

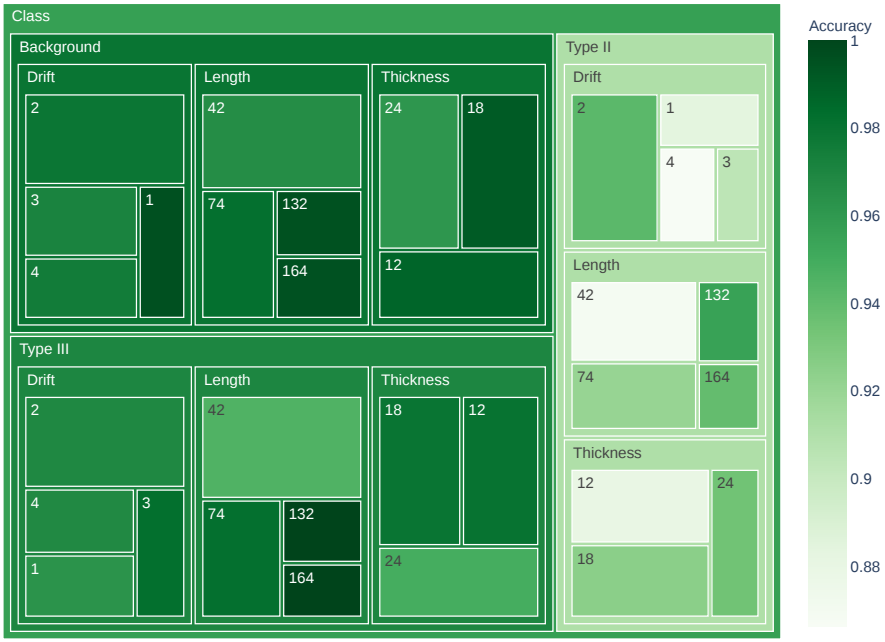


Figure 5.10: Classification accuracy by class and parameter value. Size of rectangles correspond to number of training samples (larger sizes are positioned higher and to the left). Darker shades indicate higher accuracy. Drift curve numbers correspond to those in Figure 4.12 & Table 4.1.

Longer lengths are associated with an increased accuracy rate regardless of class. For negative

samples, a longer length makes it almost certain that the sampled signal violates the drift model of type II bursts. Type II samples with a longer length may have an edge due to the increased SNR as a result of the down-scaling. The effect of down-scaling also means that longer lengths require more absolute padding to create the same perceived padding as lower lengths. It is therefore less likely for longer lengths to contain extra burst signal during padding, and this may be why we see a decrease in accuracy from a length of 132 to 164.

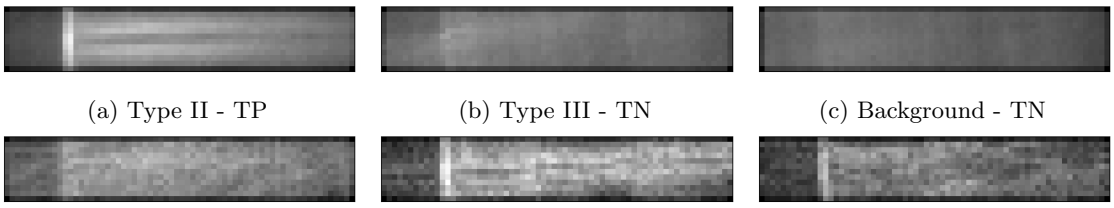
Interestingly, performance associated with the thickness parameter has a reasonably large range, with higher values being more likely to be classified correctly for type II samples. Training samples with lower thickness values seem to be more likely to capture additional signal to the left of the burst during window padding. The reason for this is unclear, but it does mean that there will on average be a weaker gradient response where the padding has taken place. As in the previously identified cases where the classifier has returned positive (Figures 5.11a, 5.11e, & 5.27), this gradient response seems to be an important feature for classifying positive samples.

The second drift curve provides a noticeable increase in accuracy for type II samples, likely due to it holding the largest share (47% of type II samples). The opposite case is true for the negatives, where the drift curve with the lowest occurrence is most accurate. This is partly expected due to the influence of hard negative mining, although the increased performance may also be an indication of the expected shape for either class. For example, type III bursts are mostly vertical, but can also present curvature at the lowest frequencies due to their decreased drift rate. Drift curves that contrast the expected curvature of type III bursts should therefore be less likely to be misclassified as positive.

ROI gradient response

Figure 5.11 compares the average gradient response for correct and incorrect classifications by class. Since the correct samples represent between 91% – 97.9% of all class samples, the corresponding gradients match the gradients seen in Figure 5.7 very closely. On the other hand, samples that have been incorrectly classified appear to deviate significantly from the average.

False positive samples appear to consistently have strong gradient responses that match the general pattern of true positives. Type III samples in particular prominently feature two horizontal strips as well as a strong response to the window padding. The two strips are not as straight as the true positives, but are still straight enough to span the entire window, and hence the corresponding type III bursts mimic the drift model of type II bursts. False positive background samples don't feature these strips as prominently, although activity can still be seen that loosely represent similar features. Furthermore, while the response to window padding isn't quite as strong as true positives or type III false positives, it is still much stronger than average. Overall, the false positive background samples don't appear to be too different to the false negatives; however, the stronger gradient intensities within the background samples seem to be the key difference that separates the two.



(a) Type II - TP (b) Type III - TN (c) Background - TN

(d) Type II - FN (e) Type III - FP (f) Background - FP

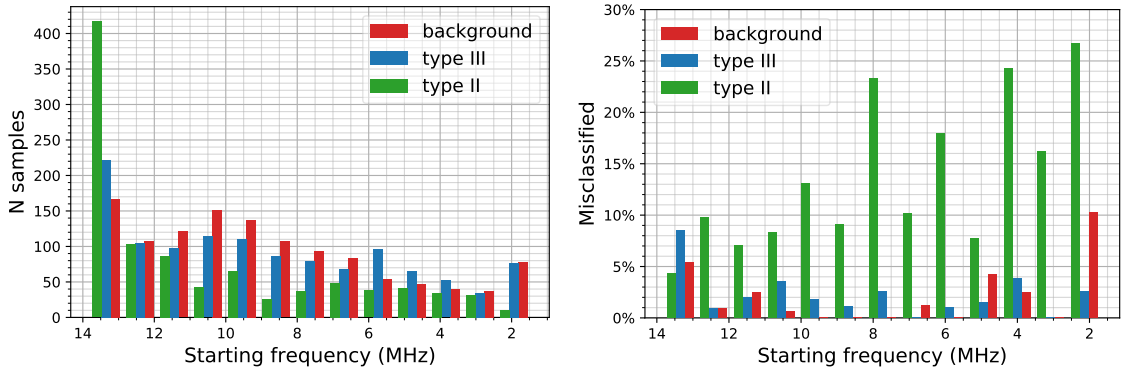
Figure 5.11: Comparison of average gradient response between correct and incorrect classifications by class. Top) Correct. Bottom) Incorrect.

ROI starting frequencies

Figure 5.12a shows the distribution of starting frequencies for our training samples. Type II bursts are significantly more likely to start at the upper boundary since the actual starting frequency may be above the data's available frequency range. The frequencies of our negatives have been uniformly sampled, although the requirement of needing enough data to fulfil the ROI's selected length ultimately results in a tailed distribution. The distribution may also be skewed towards certain frequencies due to the process of hard negative mining. Many type III bursts have been

introduced at the upper boundary due to presenting similar features to type II bursts at these frequencies. This is also true for the low frequencies, where the drift rate of type III bursts begins to decrease such that its signal presents curvature.

Figure 5.12b shows how starting frequency influences the rate of misclassification. It appears that the abundance of type II training samples at the upper frequency boundary helps to decrease the misclassification rate of these samples. Lower frequency ranges have a wide variation in misclassification rate, with the rate being as high as ∼25% for some frequency ranges. The curvature of type II bursts at lower frequencies are less likely to align with one of the drift curves, and the ability to learn the variations of misalignment is likely to suffer as a result of the low number of training samples. The number of real-world events is limited, so it is not a viable strategy to rely on new events to boost classification performance. However, at the cost of increased localisation time, it may help to increase the number of searchable ROI parameters so that existing events can contribute to a greater number of samples during our selection procedure, as well as there being an increased chance for alignment. Alternatively, instead of choosing a limited number of drift curves that aim to describe the curvature of the entire frequency range, it may help to optimise multiple sets of curves for different frequency ranges. Another option would be to optimise the selection of ROI parameters with respect to classification error, or even localisation error if using a more advanced learning algorithm. Aside from issues of drift misalignment, we also investigate how misclassification rate is influenced by length-wise ROI padding and hard negative mining.



(a) (b)

Figure 5.12: Distribution of starting frequencies for training samples by class. a) Overall distribution. b) Percentage of misclassified samples.

ROI padding

Figure 5.13 highlights a key difference between boundary and non-boundary type II samples: the gradient response at the location of padding is much stronger in the samples at the upper frequency boundary. Boundary type II samples represent ∼36% of all type II samples, so the classifier is likely to contain some bias towards the gradient response at this particular location, and hence may contribute to the disparity in misclassification rate across the different frequency ranges seen in Figure 5.12b. We have also already seen how this particular gradient response is a common theme amongst false positive samples (Figures 5.11e & 5.27), and that false negatives present the response much more weakly (Figure 5.11d).

(a) (b)

Figure 5.13: Comparison of average gradient response between boundary and non-boundary type II samples. a) Boundary samples. b) Non-boundary samples.

To evaluate the influence of padding and its resulting strong gradient more closely, we test the use of edge padding as well as no padding. We define edge padding as the case of setting the padded values to be the same as the boundary values, and thereby setting the gradient of the transition

point for each row to 0. By removing the bias of the strong gradient, our aim is to help balance the misclassification rate across starting frequencies. We recognise that edge padding may result in its own type of bias, but our aim is for it to be less significant than zero padding. Table 5.6 shows a breakdown of the change in misclassification rate by class and boundary vs non-boundary.

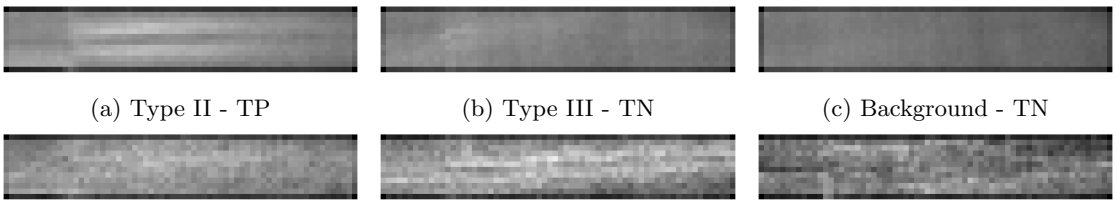
	Edge padding			No padding		
	B	NB	A	B	NB	A
Type II	+2.89	-13.47	-10.58	+6.02	-4.53	+1.49
Type III	-1.8	+13.65	+11.85	-1.8	+30.39	+28.59
Background	-2.4	+1.3	-1.1	-1.8	+17.62	+15.82
All	-1.3	+1.48	+0.17	+2.42	+43.48	+45.9

Table 5.6: Change in misclassification rate for edge padding and no padding relative to zero padding. Sample set: B=Upper boundary, NB=Non-boundary, A=All.

The removal of padding very slightly re-balances misclassification rate of type II samples, but ultimately results in the total rate to increase by 1.49 percentage points. When using edge padding, we see a much stronger effect than re-balancing: for every point increase for boundary samples, non-boundary samples see a ∼4.7 point decrease, and in total results in the misclassification rate to go down by 10.58 points. On the other hand, type III samples present the opposite effect, and consequently cancels out the reduced rate of type II misclassification. However, the use of density-based filtering means that the detection rates are likely to be low enough to prevent false positives. Conversely, the high misclassification rate at the upper boundary is likely to cause false positives, so a reduction may in fact correspond to a net benefit.

The reduced bias from edge padding may be resulting in the classifier to focus more on other types of discriminatory information, with the end result being a more relaxed decision boundary, and thus would explain the inverse association between type II and type III misclassification rates. Figure 5.14 shows that the gradient response by class and classification is almost identical to those with zero padding in Figure 5.11. One notable difference is the reduction in intensity for false negatives, which supports the notion that the decision boundary has shifted to encompass a wider feature set.

Relative to edge padding, the use of no padding consistently results in a 12 – 17 increase in misclassification rate. Figure 5.10 shows that an increased length is correlated with better accuracy, so a significant benefit of padding may be a result of the increased length. The absence of change in misclassification rate for samples at the upper boundary may be a consequence of the additional length being negligible for vertical structures such as type III bursts and calibration signals.



(a) Type II - TP (b) Type III - TN (c) Background - TN

(d) Type II - FN (e) Type III - FP (f) Background - FP

Figure 5.14: Comparison of average gradient response between correct and incorrect classifications by class using edge padding. Top) Correct. Bottom) Incorrect.

Hard negative mining

For the negative samples, frequencies with an increased sample rate from hard negative mining are experiencing high rates of misclassification. Understandably, the introduction of problem cases requires a shift in the decision boundary that may not be able to separate all of the new examples. For the most part, the non-boundary frequency ranges have a very low rate of misclassification; it is unlikely for non–type II burst signal to align with the drift model of type II bursts at these frequencies. This does however raise the issue of whether the current random sampling — which still accounts for 80% of all negative samples — is biased towards examples that are too easily

separable by the classifier. The average type III sample appears to contain little activity (see Figure 5.7b & Appendix 3 for reference), so a larger range of difficult examples may need to be sampled to enforce a stronger decision boundary.

Figure 5.15 shows preliminary results of using no padding to produce three more iterations of hard negatives. In total, we increase the size of the negative set by 18%. We see that the misclassification rate of type II samples surges to around 80% due to the classifier struggling to discriminate positive samples from the newly introduced hard negatives. This suggests that we have a weak discriminator, and that results could be improved by using more descriptive features or a more advanced classifier. During localisation, our aim is to exploit the integrated physics knowledge to produce strong detections from an aggregate of many weak detections.

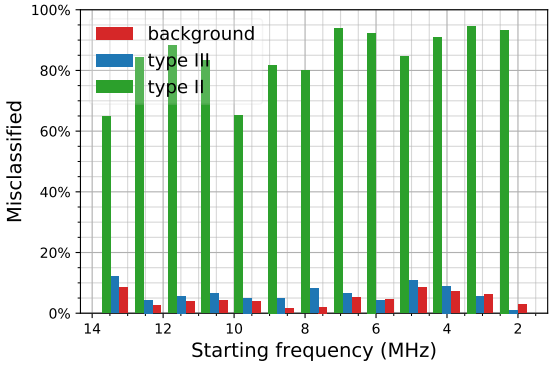


Figure 5.15: Misclassification rate after introducing more hard negative examples.

5.3 Localisation

5.3.1 Evaluation and parameter tuning criteria

We perform our localisation tests using the optimal preprocessing and feature extraction parameters from Table 5.3. We use 25-fold cross validation with the same sample grouping used in Section 5.2.1, where the samples in the test folds dictate which event windows are used for testing. For our sliding windows, we use a stride of 2 pixels for both the temporal and length axes. For simplicity, we use zero padding for boundary cases, although we recognise that edge padding is likely to be superior.

To identify true and false positives after producing a segmentation mask, we group the mask into isolated objects and check for ground truth intersections for each object. We allow any object — even a single pixel — to count as a true or false positive. In general, we can expect each object to converge to a central point as we increase the thresholding. However, we cannot assume this process will be smooth; occasionally, a single detection may split into several disconnected objects. To avoid counting each object as a unique detection, we use 5 iterations of morphological dilation on the segmentation mask prior to grouping it into objects. In effect, this results in any objects within a proximity of 10 pixels to be merged.

Recall that we use a two-stage process during segmentation: we first aim to detect burst segments using a stronger threshold that restricts the size of detections, and then we use a weaker threshold to grow the existing detections into higher quality segmentations. Any new 'detections' that arise from the lower threshold are discarded from the final segmentation. Thus, even though we only require a single pixel intersection to detect an event, it is in our best interest to ensure the initial detections are of high quality. A threshold that's too restricted may result in many burst segments to be missed, and a too weak threshold may cause segmentations to spill over the edge of the burst signal. Furthermore, a weak threshold may result in true positives to merge with false positives, and hence would inadvertently deflate the false positive rate. For these reasons, a metric such as F-score would not provide a suitable criteria for optimising the detection parameters. Recall does not indicate the quality of detections, and the correctness of precision is too unreliable. We introduce two new metrics that aim to overcome the shortcomings of both recall and precision.

The first metric, 'proportion', is a measure that quantifies how much of an event is detected in relation to its burst segments (isolated objects). For each segment, we calculate its proportion in area relative to the whole event. The proportions of all detected segments are summed to give a total score. As with recall, we only require a single intersection within a segment for that

segment to be considered detected. During the initial detection stage, our only concern is whether segments are detected, and not the intersection quality within those segments. As long as segments are detected, then the subsequent segmentation stage will improve its intersection quality. Thus, the metric can be thought of as representing the maximum potential intersection quality during segmentation. Note that this means we are not directly focused on optimising the total number of segments detected; some segments are so small that their detections would be a detriment to the overall segmentation quality. However, because we evaluate the proportion metric for each event independently, the variable burst sizes means that we do not bias the metric towards large segments only. Small segments within smaller bursts will still count towards a large relative proportion, so a high average score implies good generalisability to variable segment sizes. The metric also serves as an indirect optimisation of recall; missed events correspond to a proportion rating of 0, and thereby introduces a large penalty to the average score.

The second metric, 'relevance', is a measure of how relevant the resulting segmentation is. It is similar to the precision metric, but operates pixel-wise instead of detection-wise. However, as with the proportion metric, we are not concerned about evaluating the intersection quality at this stage. Therefore, when using Equation 5.1, TP represents the total number of ground-truth pixels within detected segments, and FP represents the number of pixels outside the ground truth. The metric's fundamental purpose is to act as a counterweight to precision; a sudden penalty will be incurred in the occurrence of a merger between true and false positive detections.

To optimise our detection parameters, we modify the F-score metric by replacing recall with the proportion and relevance metrics. As in Equation 5.4, we compute the harmonic mean of precision, relevance, and proportion. In total, we present the following metrics:

- **Precision** — Measures the likelihood of a detection being correct; see Equation 5.1.
- **FP/w** — Average number of false positives per three-hour window. Compared to precision, this metric is not biased to the total number of hours tested, and gives a more practical measure of performance in a real-world scenario.
- **Recall** — Measures the likelihood of an event being detected; see Equation 5.2.
- **Proportion (all)** — Measures the total proportion of detected segments. The mean proportion of all positive events is presented.
- **Precision (TP)** — As above, but only the true positive events are used to compute the mean.
- **Relevance** — Computes the area of all detected segments and divides this by the total number of pixels. Its intended use is to optimise the detection parameters, but we still present it as it gives an indication of the segmentation precision. The mean relevance of all positive events is presented.
- **IOU** — Measures the general segmentation accuracy. For all event windows, takes the total number of pixels that intersect with the ground truth and divides that by the total number of pixels.

$$DetectionScore = \frac{3}{Precision^{-1} + Relevance^{-1} + Proportion^{-1}}. \tag{5.4}$$

We tune three parameters for detection and segmentation: confidence threshold, length truncation, and voting threshold. Confidence refers to the probability estimate from logistic regression, length truncation is how much length is truncated from the detected ROIs, and voting threshold is the number of times a pixel has been detected. We evaluate these parameters using grid search: confidence 0.5 – 1, truncation 0% – 90%, and pixel votes 2 – 7 on a natural log scale. The score in Equation 5.4 is used as the criteria for choosing the detection parameters, and IOU is used for the segmentation parameters. We test the performance of segmentation with and without background removal and detection staging, and tune the parameters for each variation separately. To avoid testing too many parameter combinations, we evaluate the optimal length truncation for segmentation with no background removal or staging, and then use this value for the other variations. Table 5.7 shows the optimal parameters found.

	Detection	No BG removal		BG removal	
		No staging	Staging	No staging	Staging
Confidence	0.61	0.96	0.93	0.97	0.98
Voting	6.6	4.7	5	4.2	3.5
Length truncation	50%	40%	40%	40%	40%
Sigma	-	-	-	1	1
Object size	-	-	-	9	9

Table 5.7: Optimal detection and segmentation parameters.

5.3.2 Detection and segmentation

Table 5.8 shows results for detection and all variations of segmentation. When optimising for segmentation instead of detection, we see a 27.6% increase in IOU. Recall also increases by 5.7%, but at the cost of a substantial 50.8% increase in the false positive rate (FPR). Through the use of staging, we are able to sacrifice the relatively small gain in recall in favour of simultaneously achieving the lower detection-optimised FPR, as well as the higher segmentation-optimised IOU. IOU does decrease slightly after staging, which indicates that the decrease in ground truth intersections is greater than the decrease in area of false positives. In other words, the size of false positive objects are not as large as true positive objects.

We also see a decrease in the FPR, which indicates that the weaker threshold sometimes causes true positives to merge with false positives. However, we remind the reader that the ground truth has been annotated by a non-expert, and thus the merged 'false positives' may actually be burst signal excluded from the ground truth. An indication of this is that even after discarding any segments not detected in the first stage, the average detected proportion still increases, signifying that some newly arisen segment detections are in fact not being discarded. This occurs when two segments in close proximity merge into a single 'detection', and thus allowing both segments to be preserved even if only one of them was originally detected. Similarly, if two segments become merged but one of them was not annotated, then they would no longer be incorrectly considered as a false positive detection. Inevitably, the error of the ground truth is going to be reflected in the presented metrics, but the resulting trends should remain unbiased due to local errors being averaged out across all annotations.

We refine the segmentations by discarding any pixels that overlap with the background known from preprocessing. This helps to target any pixels that have spilled over the edge of burst signal, and as a result we see a 4.4% increase in IOU and a 7.9% increase in relevance. Aside from the decreased union, the ability to use a weaker threshold also helps to increase the intersection, since the risk of over-growing the segmentations is mostly removed. Without the use of staging, the weaker threshold means we see an increase in recall and proportion, but naturally this comes with the cost of an increased FPR. These benefits go away when using staging, but the benefits of the increased IOU and relevance remain with zero penalty to the FPR. Furthermore, we still see a marginal benefit to proportion due to the increased opportunities for segment mergers.

	Detection	No BG removal		BG removal	
		No staging	Staging	No staging	Staging
FP/w	0.124	0.187	0.115	0.211	**0.107**
Precision	66.0%	57.7%	67.8	55.5%	**69.4%**
Recall	72.5%	76.6%	72.5%	**78.7%**	72.5%
Proportion (all)	54.6%	61.3%	56.9%	**62.5%**	57.7%
Proportion (TP)	75.2%	**80.0%**	78.4%	79.5%	79.5%
Relevance	**85.5%**	69.8%	69.8%	77.4%	75.3%
IOU	21.7%	27.7%	27.0%	**29.2%**	28.2%

Table 5.8: Performance of detection and segmentation.

In Section 4.3.3, because we only utilise a finite amount of lengths, we are forced to create training samples where the allocated lengths are longer than the length of the burst. We first

propose potential training samples from our ground truth annotations, and then use a criterion to either accept or reject the proposals. The criterion determines how much extra length is acceptable, and thus the resulting range of extra lengths within our training samples also become present within the detected ROIs. Therefore, the resulting localisations will be poor due to the length being consistently overestimated. To correct for this, we truncate the length of all detected ROIs prior to processing them for segmentation. Since all ROIs are considered as an aggregate, a truncation equal to the average extra length within our samples should correspond to a correctly adjusted localisation.

To validate this, we test how varying the amount of truncation effects performance. Indeed, we can see in Figure 5.16 that the optimal amount of truncation for segmentation is 40%, which seems to be in agreement with the average amount of extra length added to our training samples. In Figure 5.7a, a notable decrease in gradient magnitude can be seen at around the 60% length mark of the original unpadded window. When considering detection only, the performance associated with the amount of truncation occurs roughly at an offset of 10%, where the optimal amount is 50% instead of 40%. Since we only aim for partial-coverage of burst segments during detection, it makes sense that a higher than average truncation value works well. If we match the truncation value for detection to be the same as segmentation, then we see a 34% increase in the FPR. After 20% truncation, the rate of performance decline begins to decrease, and becomes negative at 0% truncation for a performance increase. This is likely due to the influence of the proportion metric, where more segments are likely to be detected as a result of longer ROI lengths.

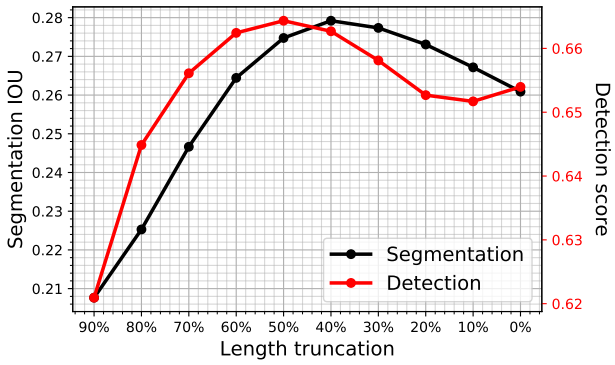
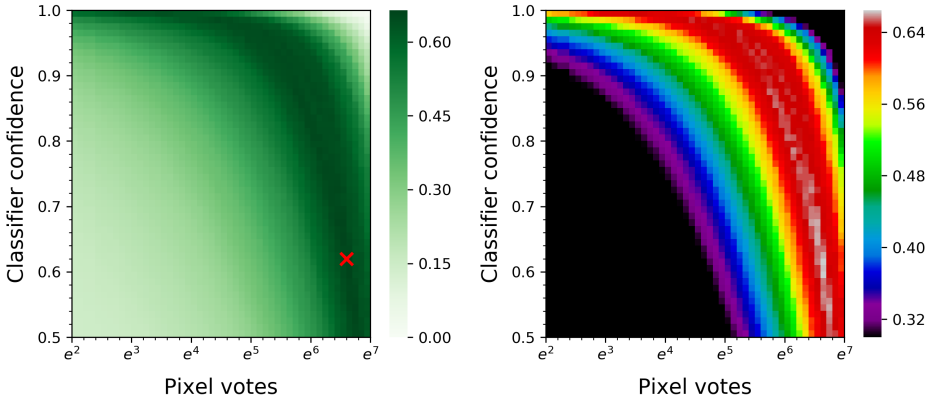


Figure 5.16: Effect of length truncation for detection (detection score; see Equation 5.4) and segmentation (IOU).

Using the optimal truncation for detection (50%), Figure 5.17a shows the effect of varying both the confidence and voting thresholds. An elbow curve can be seen wherein the two thresholds are in equilibrium: performance remains relatively static within the curve but decreases in the outwards direction. The elbow point of the curve is around 0.9 confidence and $e^{5.8}$ pixel votes, which is the point at which a small increase in either axis corresponds to a large decrease in the other. Increasing the confidence threshold is much more sensitive however; the thickness of the curve quickly narrows, but conversely remains constant when increasing the number of pixel votes. The point on the curve with the highest score is at 0.61 confidence and $e^{6.6}$ pixel votes. Figure 5.17b confirms that this is not by chance, but that there is indeed a subtle increase in score around this point.

Figure 5.18 shows the growth rate of false positives relative to recall, proportion, and relevance. The measure of false positives correspond to the mean count over all 731 three-hour windows (2193 hours in total). The rate of false positives grows much quicker than recall, and hence serves as a severe bottleneck to detection performance. At the optimal threshold, 91 false positives are detected, which equates to 0.124 false positives per 3-hour window, or 24.2 hours per false positive. This is already high, and as a result forces us to settle with a low recall of 72.5%. Recall begins by increasing rapidly, but fails to keep up with the false positive rate the moment the slope begins to decline. If we were to aim for a higher recall, such as at the elbow point of the curve and at the intersection with relevance, then we would achieve 79.1% recall and 0.22 false positives per window (161 false positives or 13.6 hours per false positive). The result is a 9.1% increase in recall, but at the cost of a substantial 77.4% increase in the false positive rate.

We can also see that the growth rate of the detected proportion is roughly proportional with



(a) (b)

Figure 5.17: Effect of classifier confidence and pixel voting on detection score (Equation 5.4). Length truncation is fixed to the optimal value (50%). a) Darker shades are better; red cross marks the optimal threshold pair. b) Same results as 5.17a, but subtle differences in score are accentuated.

recall, and thus the effect of the false positive bottleneck is exponential: a decrease in the false positive rate corresponds to a decrease in both the quantity and quality of detections. This is the inevitable drawback of identifying burst segments independently and without context of the event as a whole. We do try to gently nudge our detector to be context-aware through the use of physics-modelling, multi-segment detections, and density-based filtering — but the implicit reinforcement does not appear to be significant enough to increase the quality of detections.

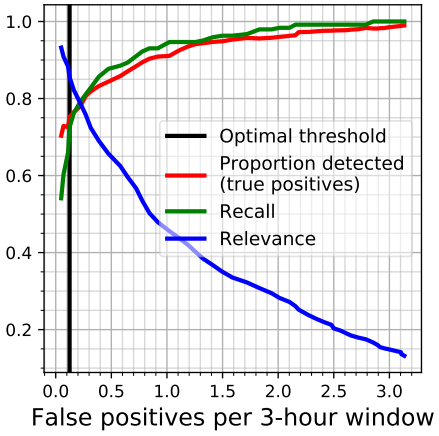


Figure 5.18: Correlation between the number of false positives and the number of true positives. Proportion and relevance metrics are also plotted for reference. The plot shows the effect of varying the number of pixel votes while keeping the confidence fixed at the optimal threshold. The optimal threshold for pixel votes is marked with a black vertical line.

Figure 5.19 shows how thresholding affects segmentation IOU when making use of staging and background removal. A similar elbow curve to the one seen for detection (Figure 5.17) is present, and its shape remains relatively constant throughout the different variations of segmentation. The distribution of IOU scores does however have some variation. Higher scores tend to be situated further towards the top of the curve when using background removal, and consequently this results in the optimal threshold to shift towards a lower voting threshold. This is an indication that the voting threshold is the primary parameter when considering the 'weakness' of the combined thresholds. We can also see that when using staging, the breadth of the curve increases, and this increases even further when combined with background removal. Another notable difference between no staging and staging is the introduction of a baseline from detection. When using very

strong thresholds, any previously detected segments remain preserved, and thus the intersection measure is much higher at these points compared to using no staging.

When adapting the method to a different instrument, we expect the selection of detection and segmentation parameters to be influenced by the physical properties of type II bursts at different wavelengths. The difference in physical properties will correspond to a difference in the selection of ROI parameters, which will in turn correspond to a difference in the resulting pixel densities. Our aim is to minimise the number of supervised examples needed, so we present the visualisations of Figures 5.17 & 5.19 as a reference for the expected performance distribution of the two parameters. The use of a small selection of annotated examples could therefore be used to loosely replicate similar trends, which can then be compared against the reference to guide the process of parameter selection.

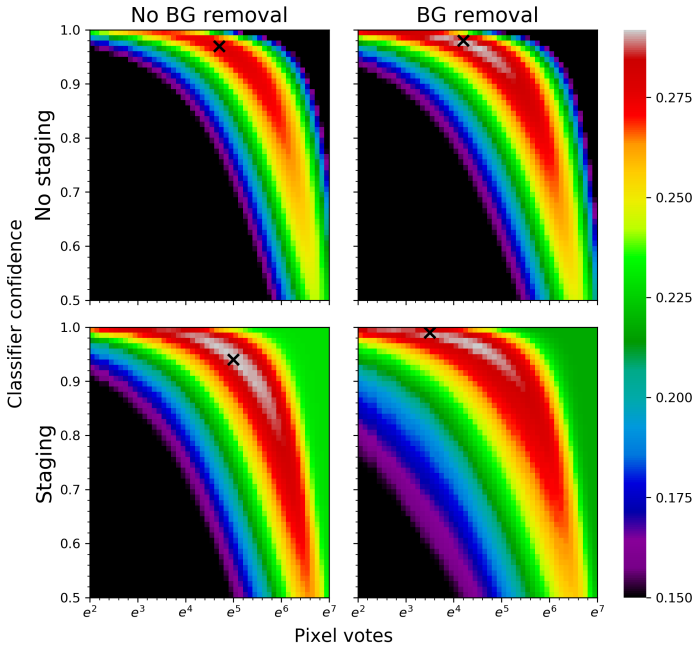


Figure 5.19: Effect of classifier confidence and pixel voting on segmentation IOU. Length truncation is fixed to the optimal value (40%). The effect of using staging or background removal is shown. Cross marks show the optimal threshold pair.

Figure 5.20 shows how varying the background removal parameters affect IOU. Performance is optimal around 1σ – 1.5σ and decreases outwards, with a higher threshold corresponding to a faster rate of performance decline. We can see from Figure 5.21a that this is a result of the relative difference in rate of change between both intersection and union measures. The two are almost inversely correlated, where the intersection's rate of change increases with the threshold, and conversely the union's rate of change decreases. The optimal threshold is where the difference between the two slopes, i.e. the ratio between the two measures, is maximum. We see that increasing the threshold has a larger affect on intersection, where the total relative change surpasses the union at 3σ. Because more true positive signal is being removed compared to false positive signal, we consequently see a greater impact on the overall IOU.

A higher threshold is always better when choosing a minimum object size, where the IOU never decreases as a result of increasing the threshold. Figure 5.21b shows that this is due to the union being affected much more than the intersection. Each measure continues to decrease, but the union's significantly quicker rate means that each step contributes to a higher ratio between the two measures. The average rate of change for the union is -0.181%, which is 6.3 times as much as the intersection's -0.029%. This supports the prior statement of false positive objects being smaller in size compared to true positive objects.

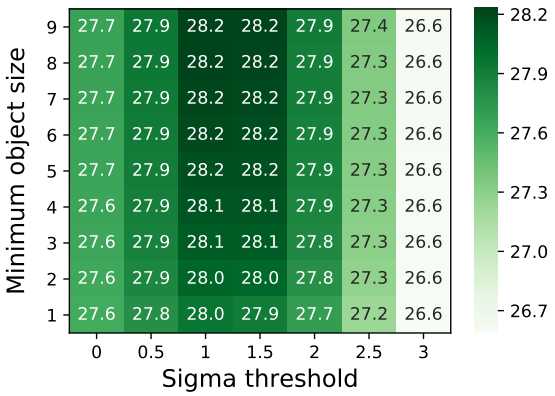
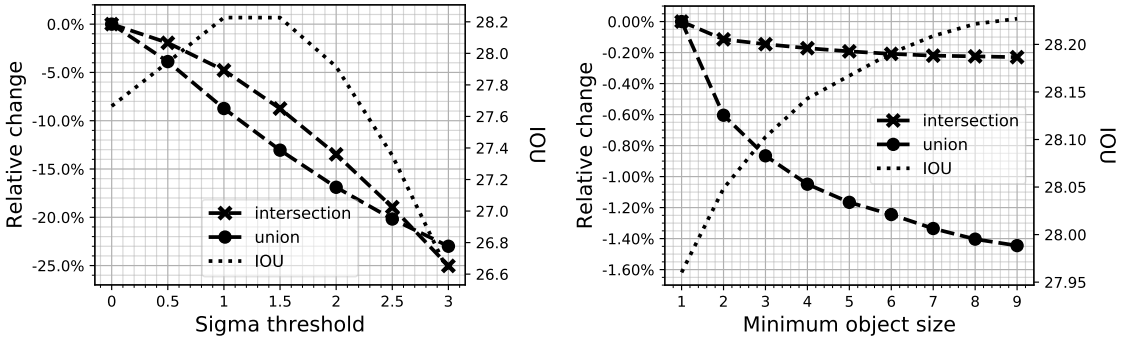


Figure 5.20: Effect of background removal parameters on segmentation IOU.



(a) (b)

Figure 5.21: Effect of background removal parameters on intersection and union magnitudes. Change is shown by the decrease in pixels for each measure. IOU is plotted for comparison.

5.3.3 Solar activity–based classifiers

We have already seen that specialised classifiers perform worse during classification, so at the risk of overfitting, we decide to optimise localisation parameters separately for each classifier. Our rationale behind this is to see whether it is possible for specialised classifiers to outperform a general classifier under the best case scenario. We utilise staging and background removal during segmentation, so in total we optimise classifier confidence and pixel votes for both detection and segmentation, as well as the sigma threshold and minimum object size for background removal. Rather than optimising separate values for length truncation, we use the values found for general localisation. Table 5.9 shows the parameters used for each specialised classifier.

	Low	Medium	High
Confidence (detection)	0.62	0.89	0.75
Voting (detection)	6.9	5.9	6.1
Confidence (segmentation)	0.92	0.93	0.93
Voting (segmentation)	3.4	4	2.5
Sigma	1	1.5	1
Object size	9	9	9

Table 5.9: Optimal detection and segmentation parameters for solar activity–based classifiers.

	Low		Medium		High		All	
	G	S	G	S	G	S	G	S
TP	**35**	22	**65**	57	**77**	73	**177**	152
FP	15	**5**	22	**17**	**41**	48	78	70
Precision	70.0%	**81.5%**	74.7%	**77.0%**	**65.3%**	60.3%	**69.4%**	68.5%
Recall	**71.4%**	44.9%	**73.0%**	64.0%	**72.6%**	68.9%	**72.5%**	62.3%
FP/w	0.104	**0.035**	0.083	**0.064**	**0.128**	0.150	0.107	**0.096**
IOU	**24.9%**	22.9%	**31.2%**	26.4%	**27.2%**	25.5%	**28.2%**	25.4%

Table 5.10: Comparison of localisation performance between general and specialised classifiers on different activity periods. G=general; S=specialised. Bold results highlight the best performing classifier for a given activity period.

Table 5.10 shows localisation results on activity subsets using both general and specialised training. The use of specialised training always results in a lower recall, and in aggregate totals to a decrease of 14.1%. However, the FPR does also decrease for low and medium activity periods, corresponding to a total decrease of 10.3%. With the decrease in recall being higher than the decrease in FPR, as well as segmentation IOU also decreasing by 9.9%, it is clear that the use of specialised classifiers reduces the overall performance of localisation. Given that no all round performance increase is observed for any particular activity period, we don't see any benefit in compositing general and specialised classifiers.

5.3.4 Failure-case analysis

False negatives by frequency

Figure 5.22 shows how the proportion of ground-truth (GT) that has been detected versus undetected changes with frequency. At the upper frequency boundary, the ratio between TP pixels and FN pixels is roughly equivalent. Afterwards, the absolute pixel counts of FNs remain relatively steady, whereas the number of TP pixels see an increase that coincides with the GT. As frequency progresses downwards, the number of GT pixels begin to decline, with the same trend of the number of TP pixels following suit. The frequency distribution of the GT is going to be representative of the patterns learned by the classifier, so it makes sense that the number of GT pixels influence the number of TP pixels. However, we do not see the same trend at the lower frequencies: the number of GT pixels begins to climb again after 7 MHz, but this time we see an increase in the number of FN pixels.

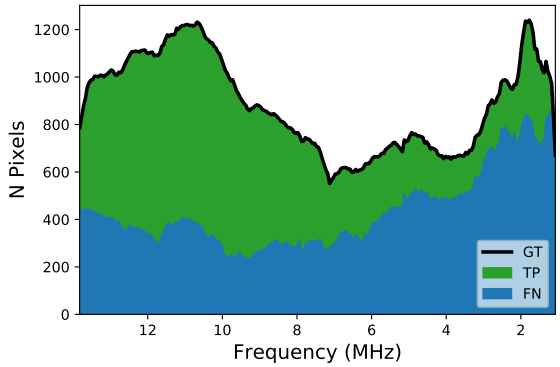


Figure 5.22: Distribution of ground-truth by frequency and segmentation output. GT=Ground-truth, TP=True positive, FN=False negative.

When comparing the frequency distribution between GT pixels in Figure 5.22 and the starting frequency of training samples in Figure 5.12a, there is clearly an imbalance between the two. There are very few training samples at the lowest frequencies even though the GT contains a large number of pixels. When selecting training samples, we impose a constraint such that the training sample must contain a certain proportion of GT signal. Training samples are therefore less likely to

start at the lower frequencies due to the artificial discontinuation of burst signal as a result of the frequency boundary. Consequently, this also gives less opportunities for signal at these frequencies to be detected, since the boundary enforces that signal must be detected at the tail end of an ROI rather than at the beginning. Given that the number of ROI intersections is a critical component in our detection methodology, the ability to identify signal at the lower frequencies will suffer as a result. Boosting the number of pixel votes at the frequency boundary may help to preserve its signal within the final segmentation.

A secondary issue is that the presence of human error may be amplifying the perceived error of the detector. For visualisation purposes, we stretch the temporal axis of the data by a factor of $\sim$2 so that structural patterns are easier to identify. This makes bursts at lower frequencies appear to be thinner, and as a consequence the relative margin of error becomes larger.

False positives by frequency and object size

After grouping segmentations into true and false positive objects, Figure 5.23 shows the occurrence rate of frequencies for objects by the class of window they were detected in. Pixels of detections within type III windows are more likely to be situated at the higher frequencies, which makes sense given the fact that the curvature of type III bursts are similar to type II bursts at these frequencies. This same trend can also be seen within background windows, indicating the presence of type III bursts that were missed by the automatic catalogue. As the curvature between the two types of bursts begins to deviate significantly, the occurrence rate of false positives within type III windows drops to 0. Within background windows, we see the opposite effect, where the occurrence rate increases significantly.

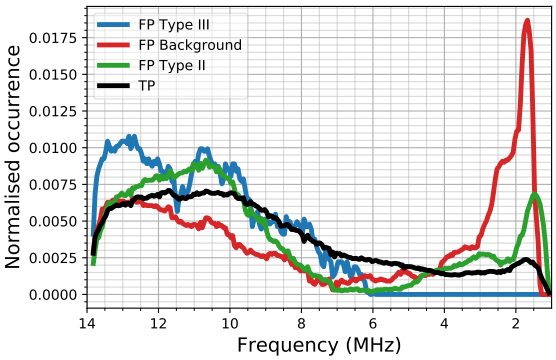


Figure 5.23: Normalised occurrence rate of frequencies by true and false positive segmented detections and window class.

Type II windows contain significantly more false positives compared to non–type II windows, with 55 false positives compared to 10 and 14 for type III and background windows, respectively. This suggests that many false positives may be a product of unlabelled true positives. However, Figure 5.23 shows that the occurrence rate of frequencies for false positives within type II windows does not follow closely to true positives, but instead seems to be a mixture of types. Figure 5.24 plots the distribution of object sizes by type to see if any similarities can be identified. From the figure alone, it is not immediately clear how the distribution of false positives within type II windows compare to other instances. On one hand, omissions during annotation are likely to be biased towards smaller, less noticeable clumps of signal. The distribution could therefore correspond to the distribution of true positives except with a scaled down range. On the other hand, the distribution is also quite similar to false positives within background windows; it is possible that the sufficiently high density of false positive detections are piggybacking off of nearby type II bursts. To better validate the causes of false positives, we take a look at some examples visually in Figure 5.25. False positives within type III windows appear to be quite distinct from other types, where the average object size is skewed towards the tail end of its narrow range.

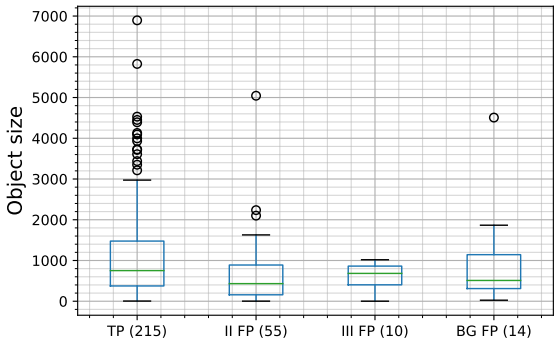


Figure 5.24: Distribution of object sizes by true and false positive detections and window class. Number of objects is shown in label brackets. Boxplots show range, interquartile range, median, and outliers. II=Type II, III=Type III, BG=Background.

False positive examples

Figure 5.25 shows examples of false positives within type II windows. We see a mixture of causes including unlabelled type II signal, type III bursts, and type IV bursts. The two most common reasons for unlabelled type II signal appears to be very weak signal, or signal that has been obscured by surrounding noise. The disparity between the number of false positives within type II windows and non–type II windows seemed to indicate that unlabelled signal would account for most of the false positives. However, the number of genuine false positives also appears to have increased. Figure 5.23 gives a notable insight to the fact that no false positives occur at the lower frequencies within type III windows, despite Figures 5.25b & 5.25c showing that the same is not true for type III bursts within type II windows. Evidently, type III bursts within the two classes of windows do not share the same characteristics. Indeed, type III bursts that precede type II bursts are a distinct class known as type III-l bursts [6, 11]. This highlights a flaw with our current approach to sampling: we only sample negatives within windows that do not contain a type II burst, but doing so fails to capture the context-dependent features of solar events associated with type II bursts. Type IV bursts are another example of this issue; their association with type II bursts becomes a restriction to their inclusion in the negative set. Type IV bursts are often falsely detected multiple times per event, and its dithering effect may be a potential cause for confusion. The inclusion of both type IV and type III-l bursts within the training set may be enough for the classifier to stop responding positively to their patterns, although as previously identified in Section 5.2.5, we may need to instead focus on using more advanced feature extractors and learning algorithms.

Figures 5.26 & 5.27 show examples of false positive detections within type III and background windows, respectively. As previously seen visually in Figure 5.25a, and analytically in Figure 5.23, high frequencies of type III bursts are a common contributor of false positives. We stop seeing false positives after a certain frequency point due to the drift trajectory of type II bursts becoming much more curved rather than vertical. Figure 5.27a shows a rare case of a type III false positive extending down to the mid frequencies. Aside from vertical structures at the higher frequencies, any structure that happens to align with the drift trajectory of type II bursts at a given frequency range is also liable to becoming falsely detected. Figures 5.26c & 5.26d show examples whose structure appears to be very similar to type II bursts, but eventually goes on to violate the drift model. We can see that as soon as this violation occurs, the response of the detector abruptly comes to a halt. Our detector is designed to look for periods of conformity to the drift model of type II bursts, but fails to consider the context of the overall morphology of the structure being detected. If we were to do so, many false positives including type III & type IV could potentially be avoided. However, the fact that type II bursts often overlap with other structures would make the avoidance of false negatives a challenging task.

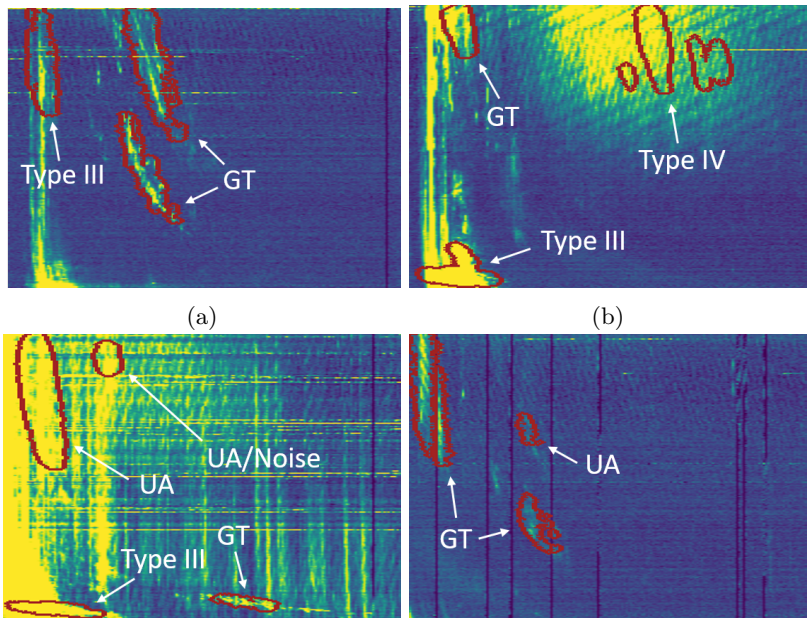


Figure 5.25: False positive detections within type II windows. GT=Positive signal included in the ground-truth, UA=Positive signal not included in the ground-truth (unannotated). False positives are seen to correspond to type III bursts at both the upper and lower frequencies, as well as a type IV burst. Some 'false positives' correspond to unlabelled signal; 5.25c shows missed signal due to overlapping with noise, and 5.25d shows weak harmonic signal that was missed during annotation.

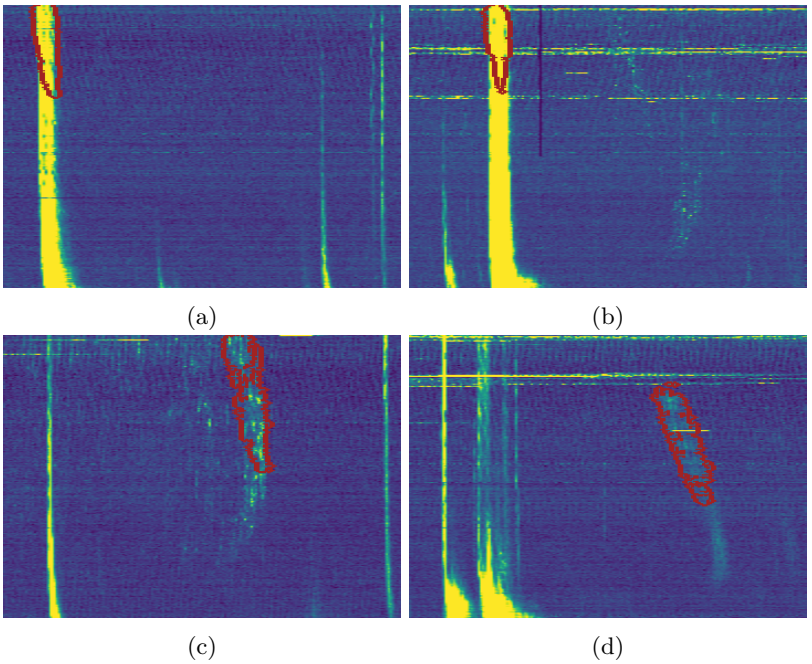


Figure 5.26: False positive detections within type III windows. 5.26a & 5.26b show how the high frequency portion of type III bursts are susceptible to being wrongly detected due to having a similar appearance to type II bursts. Similarly, 5.26c & 5.26d show examples of a large portion of signal that is neither a type II or type III burst, but mimics the drift of type II burst and hence becomes detected.

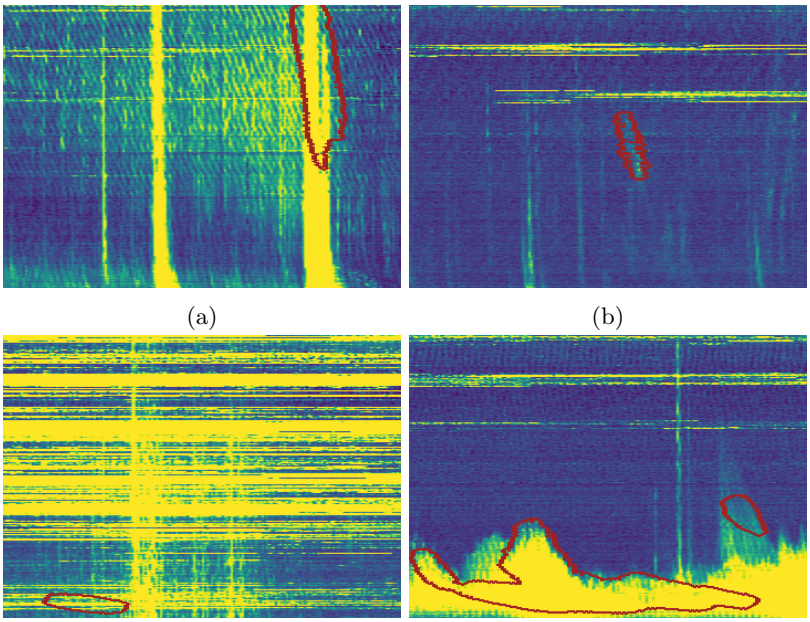


Figure 5.27: False positive detections within background windows. 5.27a shows the presence of type III bursts in background windows. 5.27b shows another example of non-burst signal that closely resembles the drift rate of type II bursts. 5.27c shows an extreme case of RFI which exhibits the abnormal behaviour of extending down to the lowest frequencies. The horizontal structure of RFI is usually drastically different to the drift of type II bursts with the exception of this unusual case. Similarly, 5.27d shows a false detection in the case of a large amount of noise situated at the lower frequencies.

Examples of undetected burst segments

Figure 5.28 shows how the use of staging can result in previously detected burst segments to become undetected. However, in the last row we can see how staging also helps to remove some false positive detections below the type II burst. We have shown in Section 5.3.2 that the trade-off between preserving true positives and discarding false positives is worth it, but it is still unsatisfying to see true positives being discarded. It is typically the smaller and weaker burst segments that are susceptible to being discarded, which tells us that these cases generally have a lower density of ROI detections, but that they are still stimulating a high enough of a response to be preserved when a weaker threshold is considered. With that said, we also see within each example cases of weak burst signal that have failed to be detected even without the use of staging.

Given that the detector is confident enough to detect some segments, it would be useful if this confidence could be used to reinforce the detection of other segments. Undetected regions that posses a moderate number of pixel votes, and are also seen to align with the drift trajectory of existing segment detections, are likely to correspond to signal that's part of the same event. The same is also true for regions that align with the $\sim$2:1 harmonic ratio of the drift trajectory. The pixel votes of these regions could therefore be boosted as a result of complying with the physics of type II bursts, which would then give them a second chance to meet the threshold. In fact, a two-way reinforcement could result in the detection of burst segments in the case where not even a single segment was previously detected.

One approach would be to aggregate the ROI parameters used to detect each object, where the number of votes for each pixel could be used to compute a weighted average. Each object would then be associated with a drift trajectory at a certain temporal position. Boosting of the pixel votes could then be a function of measures relating to the object such as confidence and distance along the drift trajectory. Going further, this would also allow for individual burst segments to be grouped into harmonic structures, where harmonics can then be grouped into complete burst events. The interdependence of these two processes — context-aware reinforcement and grouping of burst segments — could be done as an iterative procedure where each process helps out the other. Boosting pixel votes helps to find segments that can be grouped together, and grouped

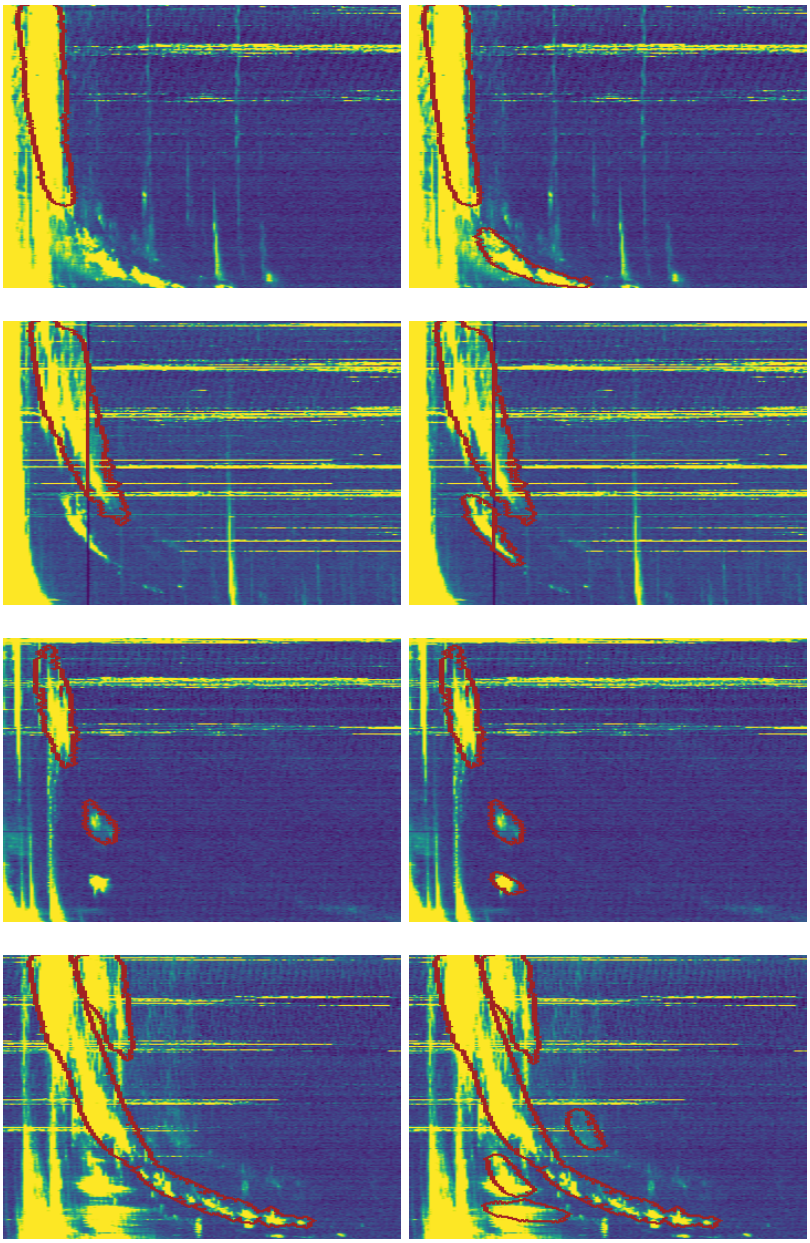


Figure 5.28: Comparison of segmentations with and without staging. Left) Staging. Right) No staging.

burst segments helps to give a more accurate model of the burst's trajectory.

6 Conclusion

6.1 Summary

We have presented a methodology which utilises prior knowledge of physics to complement the use of classical machine learning and computer vision techniques to detect and segment type II bursts. We evaluate our approach on DH type II bursts using data from Wind/WAVES, although we anticipate that our methodological design choices facilitate the application to data from other instruments. We also evaluate the usefulness of training distinct classifiers for different periods of solar activity, but found no benefit when using a dataset of our limited size.

Our noise removal focuses on the galactic background noise — an unavoidable source of noise present in all solar radio observations. We target the single constant property of the noise — its Gaussian distribution — so that robust estimations of its parameters can be made without needing to make assumptions about frequency or time dependent factors. We showcase that the use of background removal helps to improve performance for both detection and segmentation. We choose not to target sources of noise that are instrument-specific, such as calibration signals or RFI, but instead rely on the flexibility of machine learning algorithms to automatically discriminate between noise and burst signal.

Our intensity normalisation procedure is designed to make little to no assumptions about the distribution of intensity values, including distributions that may be instrument-specific or event-specific. We utilise the sigmoid function to bring a sensor's intensity values to a fixed and constant range, and then use histogram equalisation to spread out the values more evenly for enhanced contrast. We showcase that our intensity normalisation improves performance significantly, and that the use of histogram equalisation is able to maximise the benefits of sigmoid remapping regardless of the chosen parameters.

The appearance of type II bursts is heavily dependent on instrumentation: temporal and frequency resolution, spacing of frequency channels (normal vs log), and the observed frequency range are all factors that contribute to burst appearance. Our detection methodology benefits from being agnostic to instrumental spatial variances by grounding the feature representations of type II bursts with respect to their structure within time-frequency space. We achieve this through an adaptive ROI that models the curvature of bursts, followed by a straightening of the ROI using a two-dimensional coordinate transform. As a result, the appearance of bursts become much more consistent and hence the task of detection is greatly simplified. Moreover, the use of an adaptive ROI ensures that any detections are constrained to be in compliance with the structure of bursts, resulting in the prevention of false positives that correspond to impossible structures.

6.2 Future work

Our current methodology produces semantic segmentations as opposed to instance segmentations that distinguish between events and harmonic structures. Achieving the latter would be a considerable progression to this work that would enable more effective characterisations of burst parameters. In turn, this would allow insight to be gained — potentially in real-time — into the state of the Sun, and hence is a primary goal for future work. We have suggested an avenue that builds on our current work where physics knowledge is integrated into the post-processing stage for grouping together burst segments. In conjunction to this process, we also state how the process of segmentation could be improved through a feedback loop of positive reinforcement based on observed compliance with the physics of type II bursts.

Our current detector is limited in its search for ROIs due to being restricted to a predetermined set of drift trajectories. The likelihood of accurately modelling the trajectory of a burst is therefore significantly decreased, and as a consequence will also correspond to a decreased chance of detection. To improve the modelling capabilities of the ROIs, it would be beneficial to intertwine the detection stage with the stages of segmentation and segment grouping. Rather than associating each temporal position with a set of one-dimensional drift curves, each position could instead be initialised with a probability distribution of possible drift trajectories. As the detector makes new hypotheses, the probability distribution could be updated using Bayesian inference such that the focus narrows down to the true trajectory of the burst. Expanding the methodological scope further, works such as Mask-RCNN [19] have used a convolutional neural network to solve the tasks of ROI proposals, detection, and segmentation jointly by interconnecting each component under a unified network. Even when evaluating the performance of detection only, performance exceeded

the existing state-of-the-art due to being able to capture the interdependence between all three tasks. A similar approach could be considered for type II bursts where all tasks, including the addition of segment grouping and harmonic classification, could be solved jointly under a unified, physics-aware neural network. The introduction of strong contextual clues from distant burst signal could help to facilitate the detection of very weak signals, and would also allow benefits to be gained from cross-wavelength inference.

The use of a neural network also opens up opportunities to make much better use of the available data. Currently, we have only annotated 55% of events reported in the catalogue, and a further 14% of those events have not made it into our training set. Furthermore, even for the events that have been included, it is likely that a lot of information remains unused due to errors in both annotation and sampling. The use of semi-supervised learning has proven to be an effective way of achieving good performance in spite of having few ground-truth examples [21]. This is extremely useful when the cost of annotation is so high for segmentation tasks, especially in the case where domain expertise is required. Our detector provides a foundational tool for gathering cheap labels for segmentation; a weaker threshold can be used to maximise the amount of positive signal detected, where the manual removal of false positives becomes a much more efficient way to annotate events.

Self-supervised learning is another powerful technique which is able to utilise supervised learning algorithms to learn very good features from unlabelled data. A simple linear classifier achieved a 76.5% top-1 accuracy on ImageNet by using a self-supervised feature extractor [8]. Randomised supervised examples are created by generating data transformations and then learning to predict what transformation has been applied. This includes a combination of cropping, colour distortions, and Gaussian blur. Rather than explicitly integrating physics knowledge into the model, this may be an effective way to learn physics related features implicitly. Simulated data [36] could potentially allow for an easy way to do this, since it would give us access to 'fake' supervised examples without the need for labelling. Predictive tasks can then be generated very easily, since there are only two parameters to predict: presence of type II signal and intensity of signal. Being able to predict these parameters from obscured frequency channels, temporal samples, or time-frequency blocks would only be possible if a deep understanding of the behaviour of type II bursts has been learned. If this understanding has indeed been learned, then the task of adapting the problem to other instruments should be greatly simplified and may not require any supervised examples depending on the similarities between interplanetary and coronal type II bursts. A domain adaptation approach [39] may be beneficial for assisting in the transfer of knowledge.

References

[1] R Abuter, A Amorim, M Bauböck, JP Berger, H Bonnet, W Brandner, Y Clénet, VC Du Foresto, PT de Zeeuw, C Deen, et al. Detection of orbital motions near the last stable circular orbit of the massive black hole SgrA. *Astronomy & Astrophysics*, 618:L10, 2018.

[2] E Aguilar-Rodriguez, N Gopalswamy, R MacDowall, S Yashiro, and ML Kaiser. A study of the drift rate of type II radio bursts at different wavelengths. In *Solar Wind 11/SOHO 16, Connecting Sun and Heliosphere*, volume 592, page 393, 2005.

[3] Space Studies Board, National Research Council, et al. *Severe space weather events: understanding societal and economic impacts: a workshop report*. National Academies Press, 2008.

[4] X Bonnin. Type III/II radio burst automatic detection from 10 kHz to 100 MHz., 2016. Paris Observatory.

[5] J-L Bougeret, ML Kaiser, PJ Kellogg, R Manning, K Goetz, SJ Monson, N Monge, L Friel, CA Meetre, C Perche, et al. Waves: The radio and plasma wave investigation on the wind spacecraft. *Space Science Reviews*, 71(1-4):231–263, 1995.

[6] HV Cane, WC Erickson, and NP Prestage. Solar flares, type III radio bursts, coronal mass ejections, and energetic particles. *Journal of Geophysical Research: Space Physics*, 107(A10):SSH–14, 2002.

[7] HV Cane and DV Reames. Some statistics of solar radio bursts of spectral types II and IV. *The Astrophysical Journal*, 325:901–904, 1988.

[8] T Chen, S Kornblith, M Norouzi, and G Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[9] HS Dadi and GKM Pillutla. Improved face recognition rate using hog features and svm classifier. *IOSR Journal of Electronics and Communication Engineering*, 11(04):34–44, 2016.

[10] N Dalal and B Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE, 2005.

[11] RT Duffin, SM White, PS Ray, and ML Kaiser. Type III-l solar radio bursts and solar energetic particle events. In *Journal of Physics: Conference Series*, volume 642. IOP Publishing, 2015.

[12] GA Dulk, WC Erickson, R Manning, and J-L Bougeret. Calibration of low-frequency radio telescopes using the galactic background radiation. *Astronomy & Astrophysics*, 365(2):294–300, 2001.

[13] R-E Fan, K-W Chang, C-J Hsieh, X-R Wang, and C-J Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9(Aug):1871–1874, 2008.

[14] LN Garcia. Galactic Background Radiation. `https://radiojove.gsfc.nasa.gov/library/sci_briefs/galactic.html`. NASA. Accessed 2020-05-29.

[15] RC Gonzalez and RE Woods. Digital Imaging Processing. *Massachusetts: Addison-Wesley*, 1992.

[16] N Gopalswamy. Coronal mass ejections and type II radio bursts. *Geophysical monograph-american geophysical union*, 165:207, 2006.

[17] N Gopalswamy. Low-frequency radio bursts and space weather. In *URSI Asia-Pacific Radio Science Conference*, pages 471–474. IEEE, 2016.

[18] N Gopalswamy and P Mäkelä. Properties of DH Type II radio bursts and their space weather implications. In *URSI Asia-Pacific Radio Science Conference*, pages 1–4. IEEE, 2019.

[19] K He, G Gkioxari, P Dollár, and R Girshick. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017.

[20] KG Jansky. Electrical disturbances apparently of extraterrestrial origin. *Proceedings of the Institute of Radio Engineers*, 21(10):1387–1398, 1933.

[21] X Ji, JF Henriques, and A Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9865–9874, 2019.

[22] J Jones and GP Richards. Automated recognition of type III solar radio bursts using mathematical morphology. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, 2014.

[23] ML Kaiser. RAD2 RFI. `https://solar-radio.gsfc.nasa.gov/wind/examples.html`, 1995. NASA GSFC. Accessed 2020-05-29.

[24] C Kamath. *Scientific data mining: a practical perspective*, volume 112. Siam, 2009.

[25] K-L Klein, CS Matamoros, and P Zucca. Solar radio bursts as a tool for space weather forecasting. *Comptes Rendus Physique*, 19(1-2):36–42, 2018.

[26] VV Lobzin, IH Cairns, and PA Robinson. Evidence for wind-like regions, acceleration of shocks in the deep corona, and relevance of 1/f dynamic spectra to coronal type II bursts. *The Astrophysical Journal Letters*, 677(2):L129, 2008.

[27] VV Lobzin, IH Cairns, PA Robinson, G Steward, and G Patterson. Automatic recognition of type III solar radio bursts: automated radio burst identification system method and first observations. *Space Weather*, 7(4):1–12, 2009.

[28] VV Lobzin, IH Cairns, PA Robinson, G Steward, and G Patterson. Automatic recognition of coronal type II radio bursts: the automated radio burst identification system method and first observations. *The Astrophysical Journal Letters*, 710(1):L58, 2010.

[29] VV Lobzin, IH Cairns, and A Zaslavsky. Automatic recognition of type III solar radio bursts in stereo/waves data for onboard real-time and archived data processing. *Journal of Geophysical Research: Space Physics*, 119(2):742–750, 2014.

[30] LL McCready, JL Pawsey, and R Payne-Scott. Solar radiation at radio frequencies and its relation to sunspots. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 190(1022):357–375, 1947.

[31] Z Mehmood, F Abbas, T Mahmood, MA Javid, A Rehman, and T Nawaz. Content-based image retrieval based on visual words fusion versus features fusion of local and global features. *Arabian Journal for Science and Engineering*, 43(12):7265–7284, 2018.

[32] D Moran, M Lenzen, IH Cairns, and AE Steenge. How severe space weather can disrupt global supply chains. 2014.

[33] JJ Moré. The levenberg-marquardt algorithm: implementation and theory. In *Numerical Analysis*, pages 105–116. Springer, 1978.

[34] MJD Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7(2):155–162, 1964.

[35] H Salmane, R Weber, K Abed-Meraim, KL Klein, and X Bonnin. A method for the automated detection of solar radio bursts in dynamic spectra. *Journal of Space Weather and Space Climate*, 8:A43, 2018.

[36] JM Schmidt and IH Cairns. Quantitative prediction of type II solar radio emission from the sun to 1 AU. *Geophysical Research Letters*, 43(1):50–57, 2016.

[37] SILSO World Data Center. The international sunspot number. `http://www.sidc.be/silso/`. Royal Observatory of Belgium, avenue Circulaire 3, 1180 Brussels, Belgium. Accessed 2020-05-29.

[38] R Storn and K Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[39] Y Sun, E Tzeng, T Darrell, and AA Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019.

[40] S Tian, U Bhattacharya, S Lu, B Su, Q Wang, X Wei, Y Lu, and CL Tan. Multilingual scene character recognition with co-occurrence of histogram of oriented gradients. *Pattern Recognition*, 51:125–134, 2016.

[41] SM White. Solar radio bursts and space weather. *Asian Journal of Physics*, 16:189–207, 2007.

[42] SJ Wijnholds, A-J Van der Veen, F De Stefani, E La Rosa, and A Farina. Signal processing challenges for radio astronomical arrays. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5382–5386. IEEE, 2014.

Appendices

Appendix 1: List of events and detections

1997/04/01 14:00	**1**	2000/11/26 08:10	**1**	2004/06/03 16:48	**1**	2012/07/19 05:30	**1**
1997/04/07 14:30	0	2001/01/20 19:12	0	2004/06/04 07:50	**1**	2012/07/23 02:30	**1**
1997/05/12 05:15	**1**	2001/01/20 21:30	**1**	2004/06/22 22:07	**1**	2013/03/15 07:00	0
1997/09/23 21:53	**1**	2001/01/26 12:06	**1**	2004/07/23 19:00	**1**	2013/04/18 18:00	**1**
1997/11/03 05:15	**1**	2001/02/11 01:40	**1**	2004/08/08 09:15	0	2013/04/21 20:25	**1**
1997/11/03 10:30	**1**	2001/03/10 04:18	**1**	2004/10/24 03:12	**1**	2013/05/13 02:20	**1**
1997/11/04 06:00	**1**	2001/04/02 11:30	**1**	2004/11/07 16:25	**1**	2013/05/13 16:15	**1**
1997/11/06 12:20	**1**	2001/04/02 22:05	**1**	2004/11/09 17:35	**1**	2013/06/21 03:35	**1**
1997/11/27 13:30	0	2001/04/03 03:40	**1**	2004/12/08 20:05	**1**	2013/06/28 01:53	**1**
1997/12/12 22:45	0	2001/04/06 19:35	**1**	2004/12/29 16:35	**1**	2013/07/04 20:57	**1**
1998/03/29 03:05	**1**	2001/04/06 21:50	**1**	2004/12/30 23:45	**1**	2013/08/06 02:01	0
1998/04/23 06:00	**1**	2001/04/09 15:53	**1**	2005/01/04 11:20	0	2013/08/30 02:34	**1**
1998/04/27 09:20	**1**	2001/04/11 13:15	**1**	2005/01/15 06:15	**1**	2013/10/02 20:46	**1**
1998/05/02 14:15	0	2001/04/26 12:40	**1**	2005/01/20 07:15	0	2013/10/22 21:33	0
1998/05/02 17:00	0	2001/05/12 23:52	**1**	2005/05/02 22:40	0	2013/10/26 03:01	**1**
1998/05/06 08:25	0	2001/05/30 00:25	**1**	2005/05/03 00:20	0	2013/10/26 09:34	**1**
1998/05/09 03:35	0	2001/08/16 00:10	**1**	2005/06/03 12:50	**1**	2013/10/27 18:12	**1**
1998/05/11 21:40	**1**	2001/08/30 20:43	**1**	2005/06/16 20:25	0	2013/10/28 15:24	0
1998/05/19 10:00	**1**	2001/09/03 18:48	0	2005/07/09 22:15	0	2013/11/07 10:26	**1**
1998/06/16 18:20	**1**	2001/09/15 11:50	**1**	2005/08/01 14:15	0	2013/12/05 12:45	0
1998/11/02 14:00	**1**	2001/09/17 08:35	**1**	2005/08/22 01:30	**1**	2013/12/05 20:48	**1**
1998/11/06 03:00	**1**	2001/09/20 18:43	**1**	2005/08/23 15:00	0	2013/12/07 07:43	0
1998/11/07 00:20	**1**	2001/09/24 10:45	**1**	2005/08/29 11:10	0	2013/12/12 03:55	**1**
1998/11/08 11:20	**1**	2001/09/27 10:05	**1**	2005/08/31 11:40	0	2014/01/06 07:57	**1**
1998/12/18 17:50	**1**	2001/10/09 11:20	**1**	2005/09/03 03:20	**1**	2014/01/20 22:24	**1**
1999/04/24 13:50	**1**	2001/10/09 13:10	0	2006/08/16 15:45	**1**	2014/02/18 02:16	**1**
1999/05/03 05:50	**1**	2001/10/19 16:45	**1**	2006/08/26 20:40	0	2014/03/05 13:33	0
1999/06/04 07:05	**1**	2001/10/25 15:30	0	2006/11/06 10:35	**1**	2014/03/29 00:12	0
1999/06/11 11:45	**1**	2001/11/04 19:30	**1**	2006/12/06 19:00	0	2014/03/29 17:59	0
1999/06/23 05:50	**1**	2001/12/26 05:20	**1**	2007/01/25 06:55	**1**	2014/04/02 13:42	**1**
1999/06/29 19:20	0	2001/12/28 20:35	**1**	2007/12/31 01:05	**1**	2014/04/04 14:02	**1**
1999/09/03 03:00	0	2002/01/14 06:25	0	2008/03/25 19:05	0	2014/04/18 13:05	0
1999/10/14 09:10	0	2002/01/27 12:49	**1**	2010/08/01 09:20	0	2014/06/10 12:58	**1**
1999/10/17 23:27	**1**	2002/02/01 19:23	0	2010/08/18 06:05	**1**	2014/06/12 22:14	0
1999/11/16 05:17	**1**	2002/03/11 00:00	**1**	2011/01/13 09:15	**1**	2014/07/30 07:44	**1**
2000/02/12 03:55	**1**	2002/03/17 06:00	**1**	2011/01/27 12:20	**1**	2014/08/25 15:20	**1**
2000/02/17 20:42	**1**	2002/03/22 11:30	**1**	2011/02/13 17:50	**1**	2014/08/25 20:43	0
2000/03/02 13:50	**1**	2002/04/14 07:50	**1**	2011/02/15 02:10	0	2014/09/01 11:12	0
2000/04/04 15:45	0	2002/05/02 23:50	**1**	2011/03/07 14:30	**1**	2014/09/10 17:45	0
2000/04/18 15:00	**1**	2002/07/26 22:27	**1**	2011/05/29 21:10	0	2014/09/20 05:10	0
2000/05/05 16:35	**1**	2002/08/03 19:20	**1**	2011/06/02 08:00	**1**	2014/09/23 23:41	**1**
2000/05/07 21:15	0	2002/08/16 06:15	**1**	2011/06/02 12:00	**1**	2014/09/24 20:54	0
2000/05/12 23:34	**1**	2002/09/05 16:55	**1**	2011/06/04 07:00	0	2014/10/02 21:34	**1**
2000/05/15 16:47	**1**	2002/09/10 15:19	**1**	2011/08/02 06:15	**1**	2014/10/21 12:33	**1**
2000/06/02 22:00	0	2002/10/27 23:06	**1**	2011/08/04 04:15	**1**	2014/11/08 16:57	**1**
2000/06/06 15:20	**1**	2002/11/11 16:15	**1**	2011/08/09 08:20	0	2014/12/17 05:00	**1**
2000/06/06 18:45	**1**	2002/12/22 04:20	**1**	2011/09/22 11:05	**1**	2015/03/06 08:00	**1**
2000/06/10 17:15	**1**	2003/01/20 19:10	0	2011/09/25 05:30	**1**	2015/04/26 03:21	**1**
2000/06/15 19:52	**1**	2003/01/27 22:20	**1**	2011/10/21 13:15	0	2015/06/18 17:42	**1**
2000/07/22 11:45	**1**	2003/03/19 02:30	**1**	2011/11/09 13:30	**1**	2015/06/21 02:33	0
2000/08/11 11:35	0	2003/06/16 00:00	**1**	2011/12/25 18:45	**1**	2015/06/22 18:20	**1**
2000/09/12 12:00	**1**	2003/06/17 22:50	0	2012/01/02 15:00	0	2015/08/22 07:07	**1**
2000/09/19 08:45	**1**	2003/10/26 07:00	**1**	2012/01/19 15:00	**1**	2015/09/18 04:54	0
2000/10/05 22:10	**1**	2003/11/01 22:55	**1**	2012/01/27 18:30	**1**	2015/11/04 14:07	**1**
2000/10/25 09:30	0	2003/11/02 09:15	**1**	2012/03/09 04:10	0	2015/11/09 13:21	**1**
2000/11/03 18:35	0	2003/11/05 01:00	**1**	2012/03/26 23:15	0	2015/12/23 01:18	**1**
2000/11/08 23:20	**1**	2003/11/13 09:35	**1**	2012/03/27 21:45	**1**	2015/12/28 11:50	**1**
2000/11/09 01:15	**1**	2004/01/07 04:15	**1**	2012/04/09 12:20	**1**	2016/02/05 22:35	**1**
2000/11/09 16:15	**1**	2004/01/07 10:35	**1**	2012/04/15 02:30	**1**	2016/05/04 14:20	0
2000/11/23 08:16	**1**	2004/04/08 13:30	**1**	2012/07/04 17:00	**1**	2016/05/24 17:00	**1**
2000/11/24 15:25	0	2004/06/02 23:13	**1**	2012/07/17 14:40	**1**	2016/08/15 18:21	**1**

Table 1: List of events. YYYY/MM/DD HH:MM. Detected=1 | Undetected=0.

Appendix 2: Additional intensity normalisation results

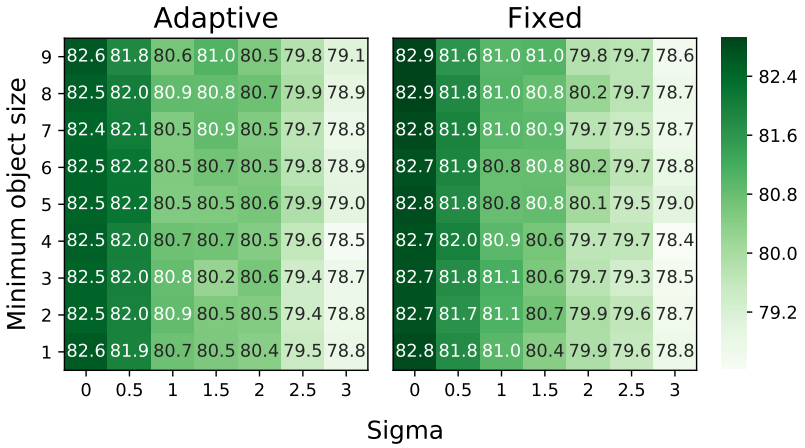


Figure 1: Linear normalisation.

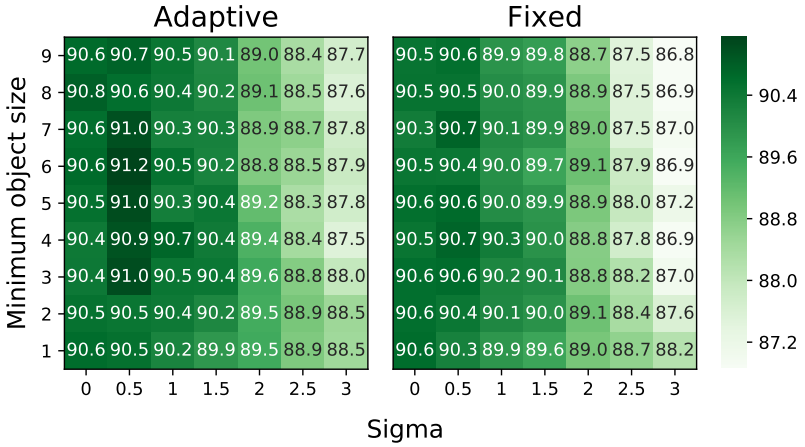


Figure 2: Sigmoid normalisation.

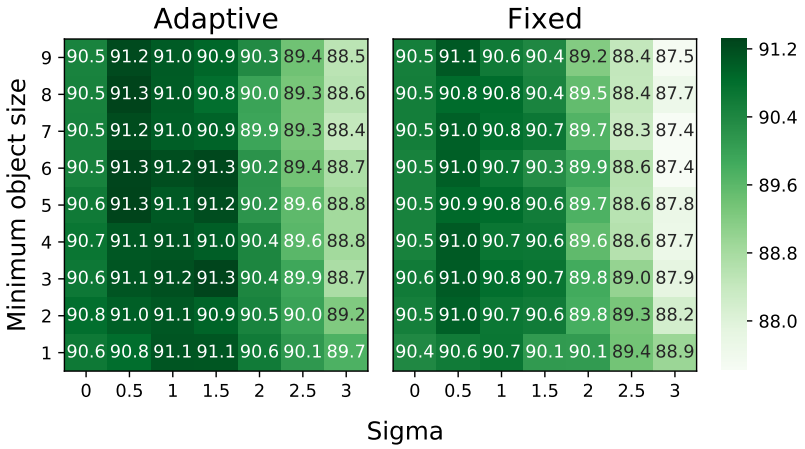


Figure 3: Sigmoid+HE normalisation with temporal scaling.

Appendix 3: Average training sample by parameter and class

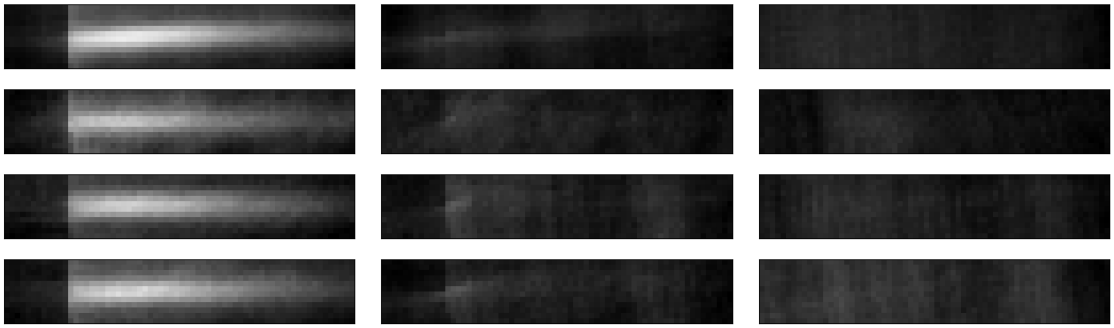


Figure 4: Drift parameter. Columns: Type II, Type III, Background. Rows: Drift 1, Drift 2, Drift 3, Drift 4. See Table 4.1 for scaling factor and power index values, and Figure 4.12 for their appearence in image space.

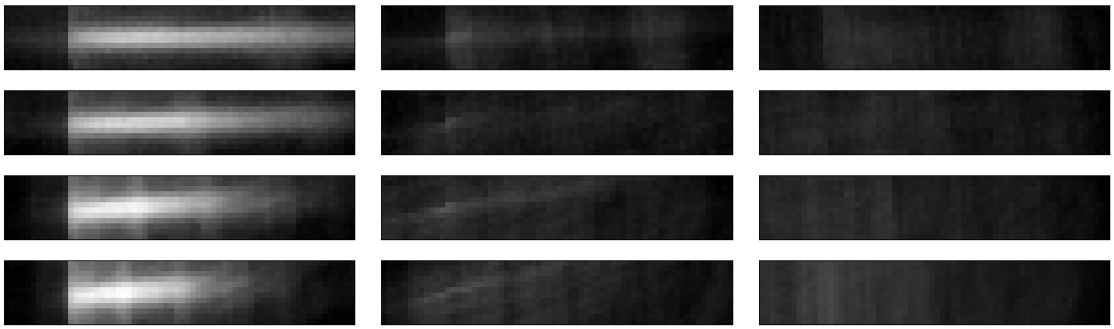


Figure 5: Length parameter. Columns: Type II, Type III, Background. Rows: 42, 74, 132, 164.

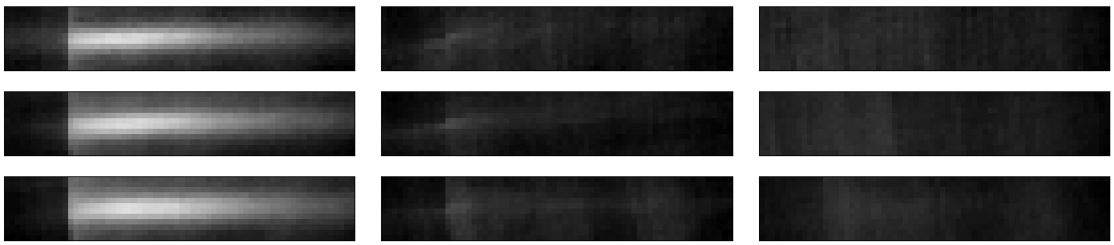


Figure 6: Thickness parameter. Columns: Type II, Type III, Background. Rows: 12, 18, 24.