

Highlights

Implementation and Performance Evaluation of the Real-Time Algorithms for Wendelstein 7-X Divertor Protection System for OP2.1

Bartłomiej Jabłoński, Aleix Puig Sitjes, Dariusz Makowski, Marcin Jakubowski, Yu Gao, Simon Fischer, Axel Winter, the Wendelstein 7-X Team

- The first real-time protection of the plasma-facing components in Wendelstein 7-X
- The primary OP2.1 system to provide safe operation with the water-cooled components
- Accelerated full-frame processing of high-resolution infrared images with GPU and CPU
- Highly sensitive overload prediction algorithm for effective machine protection
- Evaluation and analysis of the algorithm on the archived discharges from OP1.2

Implementation and Performance Evaluation of the Real-Time Algorithms for Wendelstein 7-X Divertor Protection System for OP2.1

Bartłomiej Jabłoński^{a,*}, Aleix Puig Sitjes^b, Dariusz Makowski^a, Marcin Jakubowski^b, Yu Gao^b, Simon Fischer^b, Axel Winter^b, the Wendelstein 7-X Team¹

^a*Department of Microelectronics and Information Technology, Lodz University of Technology, Wólczajska 221, Łódź, 93-005, Poland*

^b*Max Planck Institute for Plasma Physics, Wendelsteinstraße 1, Greifswald, 17491, Germany*

Abstract

A new Infrared (IR) image analysis system will be deployed for real-time divertor protection during the upcoming Operational Phase (OP) 2.1 in the Wendelstein 7-X (W7-X) stellarator. Its primary objective is to prevent thermal overloads from permanently damaging Plasma-Facing Components (PFCs), resulting in machine downtimes and repair costs. The real-time constraint for this system is 110 ms, which is the maximum allowed delay entailing the acquisition, calibration, processing, and interlock, while all processing steps have to complete within 10 ms allowing for processing longer than the acquisition time of 10 ms at 100 Hz. This paper describes the implementation, real-time processing performance and detection effectiveness of Thermal Overload Detection (TOD). The implemented and evaluated TOD system fulfils real-time constraints. It reduces the total system delay to 50 ms and provides high detection sensitivity of 0.97 for archived discharge sequences from the OP1.2 campaign. The attained acceleration is significant, i.e., a 95% and 99% decrease in runtime for the sequential Central Processing Unit (CPU) and parallel Graphics Processing Unit (GPU) implementations, respectively, compared to the initial Python prototype. For the first time, the presented results confirm the feasibility of protecting W7-X in real-time comprising fundamentals for further advanced protection and control.

Keywords:

graphics processing unit, real-time, image processing, plasma diagnostics, plasma-facing components

*Corresponding author

Email addresses: jablonskiba@dmcs.pl (Bartłomiej Jabłoński), aleix.puig.sitjes@ipp.mpg.de (Aleix Puig Sitjes), dmakow@dmcs.pl (Dariusz Makowski), marcin.jakubowski@ipp.mpg.de (Marcin Jakubowski), yu.gao@ipp.mpg.de (Yu Gao), simon.fischer@ipp.mpg.de (Simon Fischer), axel.winter@ipp.mpg.de (Axel Winter)

¹See the list of authors of Thomas Sunn Pedersen *et al.* *Nucl. Fusion*, **2022**, 62, 042022

1. Introduction

The Wendelstein 7-X (W7-X) stellarator will be protected for the first time with an autonomous real-time machine protection system based on Infrared (IR) images [1] during the upcoming Operational Phase (OP) 2.1 at the end of 2022. Its primary objective is to prevent thermal overloads that could permanently damage new water-cooled Plasma-Facing Components (PFCs), resulting in machine downtimes and significant repair costs. Machine protection systems are constrained by real-time requirements and maximum reaction time to maintain the integrity of the device [2].

The W7-X stellarator has undergone significant modifications, including its acquisition system [3, 4], since the last OP1.2 in 2018. As a result, the Field of Views (FoVs) of the cameras will be different, and one of the expected FoVs that will be used for machine protection is shown in Figure 1.

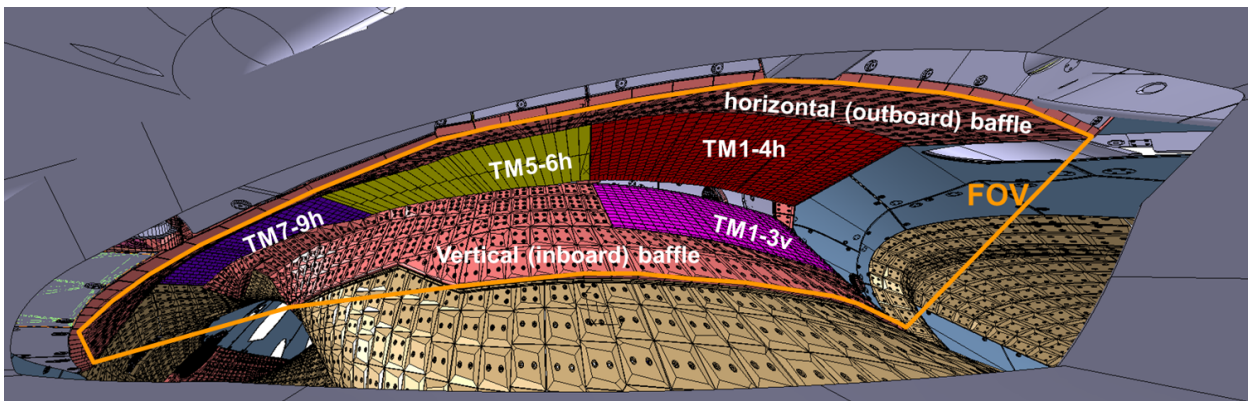


Figure 1: Simulated Field of View (FoV) for an Operational Phase (OP) 2.1 endoscope [4].

1.1. Machine Protection in Thermonuclear Fusion Devices

Machine protection with IR images is a mandatory system to provide a safe high-power operation for thermonuclear fusion devices. Therefore, some devices have already been protected with machine protection systems. At W Environment in Steady-state Tokamak (WEST), a real-time Wall Monitoring System (WMS) uses IR images from six FoVs to monitor the surface temperature in pre-defined Regions of Interest (ROIs) with fixed alarm and temperature thresholds [5]. Not only does WMS protect the PFCs from damage, but also it modulates heating power to control the surface temperature when it exceeds the corresponding ROI alarm threshold. As a result, the surface temperature decreases, and a pulse can be continued without termination. At Joint European Torus (JET), the maximum temperature in each ROI is sent to Vessel Thermal Map (VTM), and if the surface temperature exceeds the established fraction of the temperature threshold, i.e., 90%, the alarm is triggered [6]. At Axially Symmetric Divertor Experiment Upgrade (ASDEX Upgrade), the Video Real-Time (VRT) safety system, which is based on Visible Spectrum (VIS) cameras, incorporates three algorithms for machine protection in pre-defined ROIs [7]. The first one calculates

a normalised integrated intensity, the second one identifies overheating hot spots, and the third one discriminates between moving and stationary particles. As a result, each algorithm produces a numerical value per ROI, which is compared against a pre-configured threshold. Then the status is sent to Discharge Control System (DCS), which might terminate the running experiment. In addition, machine protection at ASDEX Upgrade is supplemented with the surface temperature monitoring of in-vessel components using IR systems [8]. At W7-X, it is planned to incorporate a dynamic per-pixel surface temperature threshold calculation based on the heat flux estimation [9]. The implementation, optimisation and evaluation of the W7-X system are the subjects of this paper.

Furthermore, future high-power tokamaks will be equipped with IR machine protection systems. At ITER, the first device to demonstrate the net energy gain in Magnetic Confinement Fusion (MCF), it was identified that the surface temperature and the maximum surface temperature measurements have primary roles in machine protection [10]. At DEMONstration Power Plant (DEMO), the first device to become a fusion power plant, IR imaging is classified as a critical diagnostic for machine protection [11].

Even though IR-based machine protection systems are essential for fusion devices to operate safely, these systems are facing various unsolved challenges [12]. Reflections in IR images introduce parasitic signals, which affect the observed surface temperature and might trigger false alarms if not removed. The true surface temperature is recovered offline from experimental IR images by simulating IR images with a synthetic diagnostic and solving the inverse problem to filter out reflections [13]. Moreover, new advanced solutions include approaches based on Artificial Intelligence (AI) [14] to detect, track and classify thermal events so that their characteristics are incorporated into the machine protection decision process.

1.2. Research Objective

In contrast to the IR machine protection systems incorporated in other fusion devices, the W7-X system processes full frames instead of ROIs and triggers the alarm at a dynamically adjusted temperature threshold instead of a fixed threshold. It significantly increases computational complexity since more pixels are processed with a higher number of operations. The previous studies verified the possibility of processing full W7-X IR images [15, 16]. Moreover, it was illustrated that General-Purpose Computing on Graphics Processing Units (GPGPU) techniques provide notable acceleration in this application.

This paper describes the implementation, real-time processing performance and detection effectiveness of Thermal Overload Detection (TOD), addressing real-time image processing algorithms for pre-processing and detecting overloads in W7-X IR images for the machine protection system that will be deployed for the upcoming OP2.1 campaign [9].

2. Methods

The design and theoretical justification of TOD, which is implemented and evaluated in this paper, have already been described in [9]. The following paragraphs summarise the description and provide implementation details. TOD consumes a calibrated IR image

of PFCs provided by a calibration system and produces a binary alarm signal for a Fast Interlock System (FIS) [17, 18], as illustrated in Figure 2. The processing is supplemented with a scene model prepared for each FoV. It provides additional constant pixel-wise data for processing algorithms.

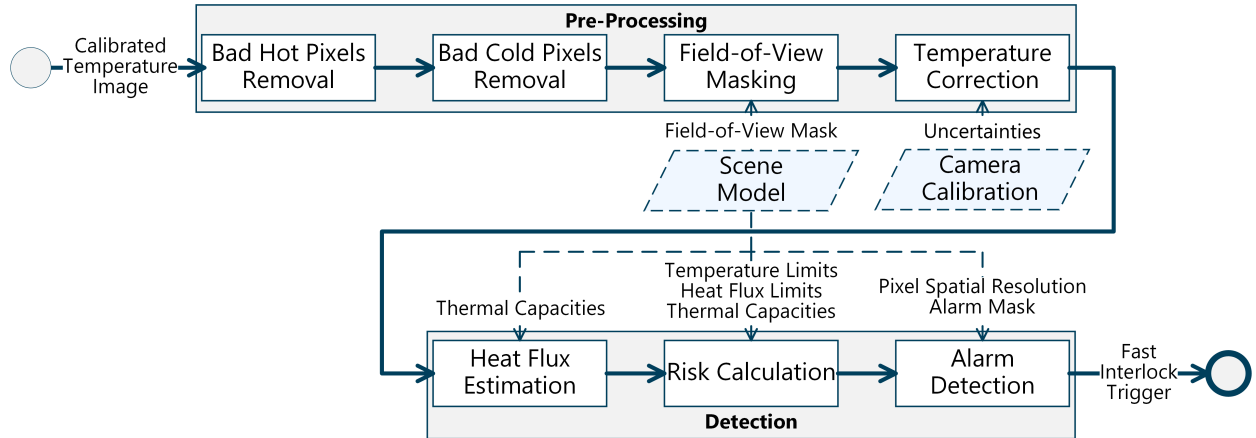


Figure 2: Thermal Overload Detection (TOD) processing pipeline. The scene model and camera calibration provide additional image data with equal dimensions to the input images, e.g., each pixel has the corresponding temperature limit.

Calibration of a raw acquired image, which is not a subject of this paper, entails Non-Uniformity Correction (NUC), Bad Pixel Correction (BPC) and temperature conversion according to Planck’s Law [9]. In addition, the calibration procedure corrects temperature offset and drift, which occur as uncooled IR sensors heat up due to stray radiation. However, it does not apply to new OP2.1 actively cooled IR sensors. This stage produces a calibrated image and an additional image with per-pixel calibration uncertainties.

The pre-processing stage applies image processing morphological operators to filter out unforeseen bad pixels that arose during the operation or due to neutrons hitting the camera sensor in deuterium plasmas. The morphological opening [19] is applied to remove bad hot pixels, and the morphological closing by reconstruction [20] is applied to remove bad cold pixels. The coldest pixels cannot trigger the alarm, but they might cause an overestimation of heat flux when they suddenly heat up. All the morphological operators use 3×3 kernels and 8-way connectivity. This kernel size offers filtering capabilities yet does not remove relevant information for machine protection, e.g., small overloads. It is a requirement of the optical system that the resolution must be at least three pixels per target tile. Therefore, filters do not remove the entire overload if the whole tile is affected. The next pre-processing step adds the upper bound of the sensor noise level and per-pixel calibration uncertainties. The last pre-processing operation averages the surface temperature using a sliding window over the current and two previous frames, i.e., a sliding window width equals three. The temporal temperature filter reduces fluctuations between consecutive frames, which would overestimate heat flux. Consequently, it removes fast surface transients that are not dangerous for the PFCs because they do not affect the bulk of the tiles. Nonetheless, it also

introduces an additional delay to the system; therefore, the sliding window width must be kept small. The detection stage dynamically adjusts pixel-wise temperature thresholds based on the estimated heat flux magnitude to anticipate overloads compensating for the total system delay. Eventually, isolated detections are filtered out in regions with high spatial resolution, e.g., the low-iota target (TM1-4h) has a higher per-pixel resolution than the high-iota target (TM7-9h) (see Figure 1).

If any overload is detected, FIS is triggered to terminate a discharge by stopping the heating systems.

Suitable software and hardware are essential in order to fulfil the following real-time requirements established for the W7-X protection system:

- 110 ms – the maximum allowed delay entailing the image acquisition, image calibration, TOD processing, FIS trigger and heating shutdown;
- 10 ms – the maximum runtime allowed per process to enable pipelining so that each image is processed by a separate thread in a queue and completes before the next frame is acquired. Pipelining allows for processing longer than the acquisition time, which is 10 ms at 100 Hz (see Figure 3).

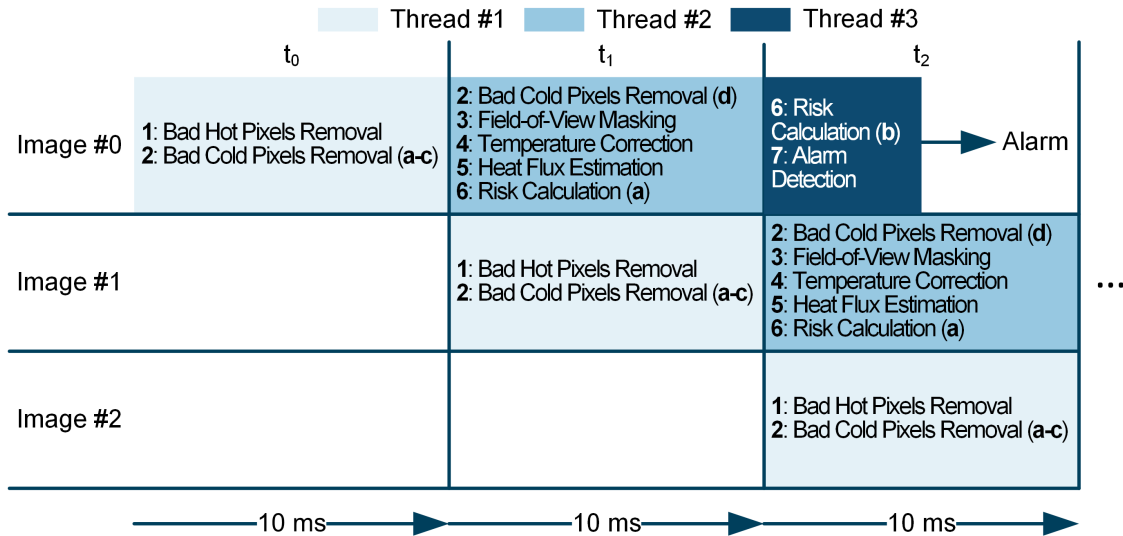


Figure 3: Image processing distribution. Processing steps correspond to those shown in Figure 2, yet they are combined or split according to their runtime to fit within a 10 ms processing limit, except for the last thread, which might finish faster. The sub-processing step division is shown in Figure 6.

2.1. Machine Protection Software

The primary characteristics of the implemented real-time algorithms for TOD are:

- The code is implemented in pure C++20 with Standard Template Library (STL) for full control over processing and resources, i.e., no external dependencies are interfering with the real-time aspects of the code;

- All input, output and intermediate buffers are pre-allocated on the stack so that no dynamic memory allocation is utilised;
- The total processing runtime is mostly invariant to the image content, i.e., the average performance for the randomised and real images is similar, and the standard deviation of the runtime between benchmark iterations is insignificant (<0.05 ms);
- Constant parameters and basic calculations are implemented with C++ templates and compile-time constant expressions. For instance, image dimensions and kernel sizes are templated to enable more compiler optimisations and compile time calculations. Moreover, it enables template specialisation to optimise algorithm implementation without incurring runtime overhead [21], e.g., the morphological erosion and dilation have a specialised implementation for the 3×3 kernel that unrolls loops and eliminates border condition checks.

The most computationally intensive computations in the pipeline are morphological image processing operations that are non-linear transformations analysing geometrical structures [19]. The morphological closing by reconstruction [20] is the most complex one in the pipeline. It is applied to remove bad cold pixels. The algorithm removes small holes, i.e., cold pixels, while preserving the temperature distribution shape. In effect, hot pixels from PFCs with a high-temperature limit, e.g., divertors, are not propagated to PFCs with a low-temperature limit, e.g., baffles, and not-cold pixels remain close to their original value (see Figure 4).

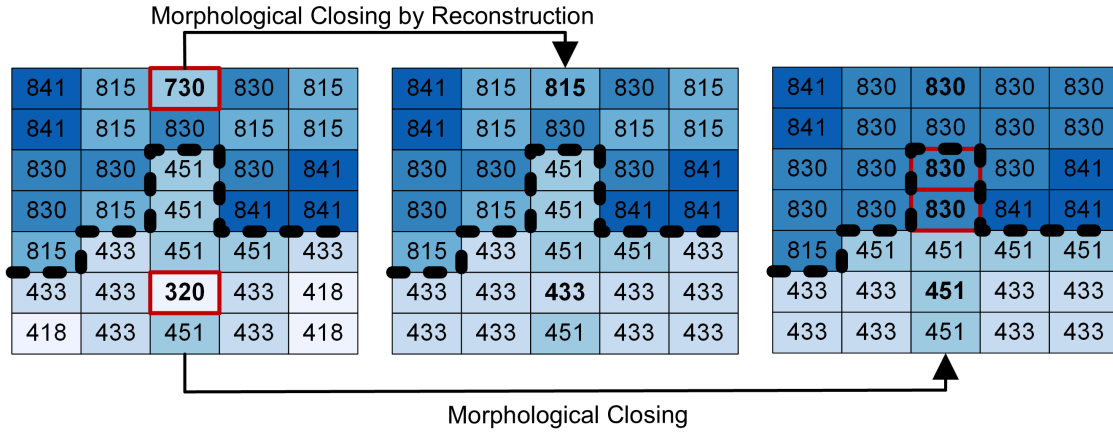


Figure 4: Comparison of bad cold pixel removal using the morphological closing by reconstruction and the morphological closing in a 5×7 surface temperature image. The dotted line is a boundary between the divertor (upper), with a temperature limit of 1473 K, and the baffle (lower), with a temperature limit of 673 K. Both methods replaced the coldest pixels, i.e., 730 K and 320 K; however, the morphological closing propagated hot divertor pixels to the baffle resulting in the false alarm, and significantly altered not-cold pixels. The shape of the component boundaries is extracted from the temperature limit image for the discharge sequence 20181017.041 (AEF40). The upper left pixel maps to the pixel at coordinates (x: 246; y: 391) of the temperature limit image.

In order to compute the morphological reconstruction in real-time, it was necessary to determine the optimal method. The standard algorithm is iterative, which is unsuitable for real-time processing. It was observed that the number of iterations required to converge exceeds 300 for processed images. Therefore, the sequential reconstruction, queue of pixels reconstruction, Fast-Hybrid (FH) reconstruction [20] and downhill filter reconstruction [22] algorithms were implemented and analysed. It was observed that the morphological reconstruction complexity is reduced in the TOD setting. It is because bad hot pixel removal, the first pipeline step, effectively reduces the number of pixels to reconstruct in the second step. Moreover, the FH reconstruction offers the highest stable performance regardless of the image content when used for the morphological closing by reconstruction in the TOD setting. An additional optimisation applied for the Central Processing Unit (CPU) morphological reconstruction is image padding with the value of 65 535, which corresponds to the maximum value for the unsigned 16-bit integer, i.e., the native data type of acquired infrared images. This optimisation effectively makes border pixels transparent for the algorithm as the border value is greater than any temperature value. As a consequence, the runtime is reduced since border and neighbourhood condition checks are no longer required (see Figure 5).

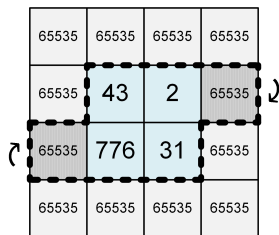


Figure 5: Padding example for a 2×2 image. The dashed outline marks the pixels that are processed. The arrows indicate the pixels for which access to its neighbour is wrapped since the image is stored as a continuous linear array.

A parallel FH reconstruction algorithm for Graphics Processing Units (GPUs) was proposed in [23]. In TOD, the first phase, including row-wise and column-wise scans, is removed because it was a bottleneck for 1280×1024 images. Consequently, the algorithm is simplified to the parallel queue of pixels reconstruction consisting of initialisation and wavefront propagation phases. The pixel queues for pixel indices requiring processing are stored in the global GPU memory as linear arrays [24]. The abovementioned padding optimisation is not used in this implementation as each thread can efficiently check the border and neighbourhood conditions in parallel. The pseudocode is shown in Appendix A in Listing 1.

The second longest runtime originates from the morphological erosion/dilation operations, which are applied four times per frame. They are applied twice to remove bad hot pixels with the morphological opening [19], once in the abovementioned morphological closing by reconstruction and once to filter out small risk clusters in regions with a high pixel resolution in the alarm detection step. The optimal performance was attained with a separable filter for the 3×3 kernel, where the two-dimensional filter is separated in a one-dimensional row-wise pass followed by a column-wise pass. No performance gain was

observed for the van Herk/Gil-Werman (vHGW) algorithm [25].

The remaining algorithms are pixel-wise computations. In the CPU implementation, the image is iterated multiple times to apply each individual operation to facilitate Single Instruction, Multiple Data (SIMD) vectorisation when accessing consecutive pixels, and the cache-friendly compute and access pattern. In the GPU implementation, all operations are applied at once to the pixel by each thread.

2.2. Machine Protection Hardware

During OP2.1, TOD will be run for each individual IR camera on a dedicated Fast Control Station (FCS) [26], which specification is shown in Table 1.

Feature	Description
CPU	24-core AMD EPYC 7402P @ 2.8 GHz
GPU	2560-core NVIDIA Tesla T4 16 GB GDDR6
RAM	4× 16 GB 64-bit DDR4 @ 3200 MT/s
PCIe	Gen 3 ×16
OS	Real-time Ubuntu 20.04.4 LTS

Table 1: Technical specification of Fast Control Station (FCS).

IR images will be acquired by eight IRCam Caleo 768kL and four SCD Hercules cameras with a resolution of 1024×768 and 1280×1024 , respectively. In the following benchmarks and evaluations, the worst-case scenario is considered, i.e., 16-bit images of resolution 1280×1024 . All ten W7-X divertor units will be covered by TOD.

3. Results

The results entail the benchmarks of the optimised processing pipeline and the evaluation of detection effectiveness with selected performance metrics relevant to machine protection aspects. Google Benchmark (<https://github.com/google/benchmark>) library is used for benchmarking. The average runtime is measured for images from the archived OP1.2 discharge sequences. For statistical validity, additional measurements using randomised images were collected. No significant average runtime deviations were observed, yet randomised unstructured images seem to be slower to process. Therefore, the reported times are for randomised 1280×1024 images to cover the worst case.

3.1. Real-Time Performance

The W7-X team initially implemented the pipeline in Python to evaluate its applicability for machine protection. The attained performance of this prototype is 404.21 ms. It was implemented using the scikit-image (<https://scikit-image.org/>) library for image processing and the NumPy (<https://numpy.org/>) library for statistics and the multidimensional data container. The performance of the prototype serves as a baseline for further optimised implementations.

FCSs will orchestrate TOD execution via the real-time CPU C++ framework. Therefore, the first optimised implementation is a sequential CPU one, which utilises a single thread per algorithm. The attained performance of the optimised sequential CPU C++ implementation is 21.74 ms, i.e., a 95% decrease in runtime compared to the Python implementation, as visualised in Figure 6.

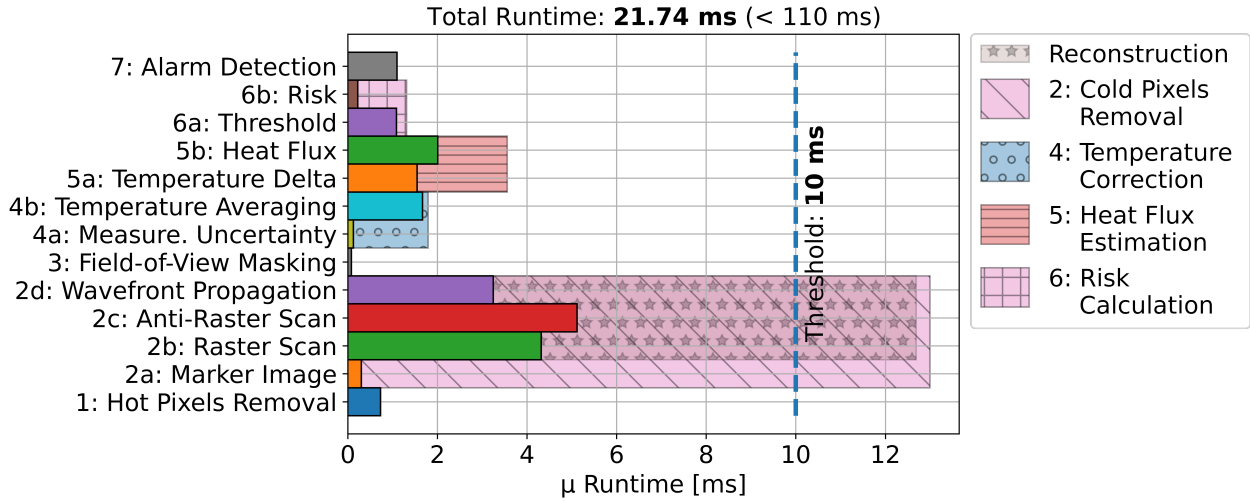


Figure 6: TOD sequential Central Processing Unit (CPU) processing performance. The total runtime is measured for all steps executed in a sequence.

According to the approach shown in Figure 3, three threads are required to pipeline the TOD processing at 100 Hz acquisition frequency. The steps might be further divided to fit within the processing limit if additional overhead is incurred by the final system, e.g., by a framework orchestrating threads.

TOD is also implemented in Compute Unified Device Architecture (CUDA) to run on the FCS's GPU. The attained processing runtime is 2.51 ms, i.e., an 88% decrease in runtime compared to the sequential CPU implementation. The bad hot and cold pixel removal are computed in 1.68 ms, and the remaining processing is computed in 0.444 ms. The measured runtime includes all transfer overheads equal to 0.425 ms. It is the time necessary to transfer the 16-bit 1280×1024 IR image and calibration uncertainty image via Peripheral Component Interconnect Express (PCIe) to the GPU and return a single 32-bit value corresponding to the binary alarm. The transfers are optimised using page-locked memory buffers on the host side to allow for efficient Direct Memory Access (DMA) transfers between the host and the device. The performance comparison between three implementations is shown in Figure 7.

Although the parallel GPU implementation provides superior performance, the sequential CPU implementation will be used during OP2.1 as it is a requirement imposed by the device operation group. Considering the rounded-up sequential CPU processing performance attained, a total system delay of 50 ms is derived, as shown in Figure 8. This delay will be used to evaluate detection effectiveness.

The parallel GPU implementation of TOD could be executed on every frame without pipelining as it is significantly faster than the acquisition rate of 100 Hz. In addition, the

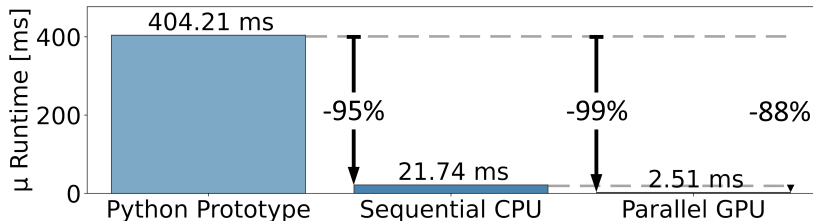


Figure 7: Runtime comparison between TOD implementations.

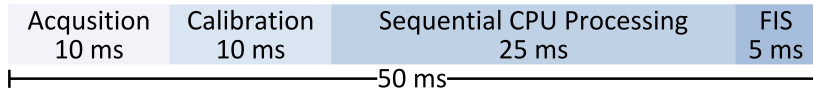


Figure 8: Breakdown of a total machine protection system delay.

remaining time for the processing allows for fusing the processing with the calibration, which consists of embarrassingly parallel computations. As a result, it would reduce the overall system delay to approximately 20 ms.

3.2. Detection Effectiveness Evaluation

During the previous experimental campaign OP1.2, no autonomous protection system was in operation, and the inertially-cooled (uncooled) test divertors were affected by overloads. However, Test Divertor Units (TDUs) could not have been significantly damaged even if the incident plasma heat flux was above 10 MW m^{-2} [27]. During OP2.1, PFCs will be water-cooled and become vulnerable to overloads. TOD is evaluated on the archived OP1.2 discharges based on a difference, called anticipation time, between detection and overload timestamps. The overload occurs when the PFC temperature limit is exceeded in the corrected temperature image. Prediction outcomes are defined in Table 2.

Detection Result	Definition
True Positive (TP)	The alarm is triggered within $[-1000, -50]$ ms before the overload
True Negative (TN)	NEITHER the alarm NOR the overload occurs
False Positive (FP)	EITHER the alarm is triggered within $(-\infty, -1000)$ ms before the overload OR the alarm is triggered when no overload occurs
False Negative (FN)	The alarm is triggered within $(-50, \infty)$ ms before the overload, i.e., it is too late to compensate for the system delay

Table 2: Definitions of prediction outcomes for TOD in the archived OP1.2 discharges. 50 ms is the system delay, and 1000 ms is the arbitrarily selected limit, which determines too early detection.

Data from all modules throughout the entire OP1.2, totalling 1419 discharges corresponding to 12 074 sequences (19 447 678 images), was processed. Discharge sequences that start overloaded due to camera synchronisation problems, those which include corrupted images, or those where the camera shutter failed to open were not included in the evaluation dataset. A detection confusion matrix is shown in Figure 9, and a histogram of relative alarm times, including TP and FN and excluding FP for clarity, is shown in Figure 10.

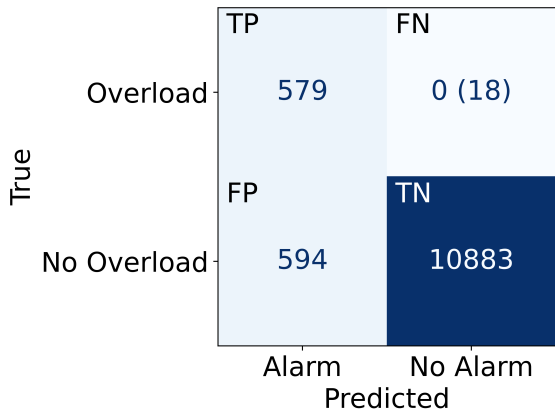


Figure 9: Confusion matrix. The final False Negative (FN) count is zero since all the late alarms are justified.

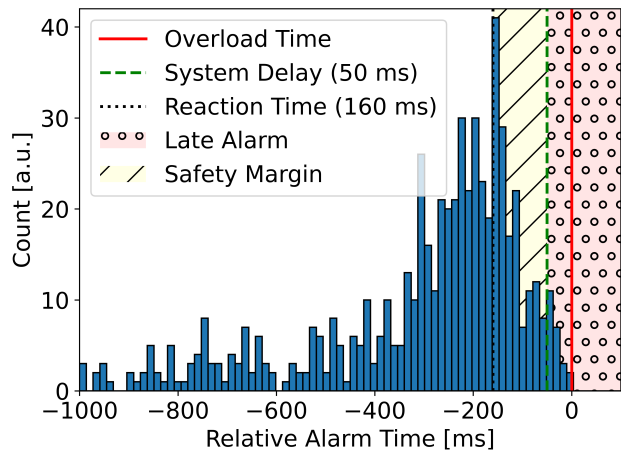


Figure 10: Alarm time histogram relative to the overload timestamp. The reaction time is the point in time at which we make the prediction, and the safety margin enables more prediction attempts to reduce uncertainty.

The most relevant metric for this machine protection system is the maximisation of sensitivity (see Equation 1), followed by the minimisation of False Positive Rate (FPR) (see Equation 2). For the GPU implementation, sensitivity is equal to 0.997 and FPR is equal to 0.042.

$$sensitivity = \frac{TP}{TP + FN} = \frac{579}{579 + 18} = 0.970 \quad (1)$$

$$FPR = 1 - specificity = \frac{FP}{FP + TN} = \frac{594}{594 + 10883} = 0.052 \quad (2)$$

3.3. Baseline Comparison with the Fixed Threshold Approach

The detection effectiveness was evaluated for the fixed temperature threshold approach for a baseline comparison. This approach does not require the bad cold pixel removal, average temporal filter, heat flux estimation and dynamic threshold calculation, i.e., steps 2a, 2b, 2c, 2d, 4b, 5a, 5b, and 6a shown in Figure 6. Consequently, the TOD processing time is reduced to 5 ms, and the total system delay is reduced to 30 ms when executed on a CPU and to 20 ms when executed on a GPU. The shorter processing of the fixed threshold on a GPU than the dynamic threshold on a GPU is insignificant, as the anticipation time is multiple of 10 ms (frame rate). Therefore, both are rounded up to 20 ms, as the remaining acquisition, GPU calibration, and FIS delays account for more than 15 ms. The alarm is triggered if the corrected temperature is greater than or equal to the fixed percentage of the PFC temperature limit. Sensitivity and FPR for four different threshold levels are shown in Table 3.

Threshold	Delay = 30 ms (CPU)		Delay = 20 ms (GPU)	
	Sensitivity	FPR	Sensitivity	FPR
80% of the PFC limit	0.990	0.263	0.997	0.263
85% of the PFC limit	0.984	0.166	0.995	0.166
90% of the PFC limit	0.963	0.095	0.994	0.095
95% of the PFC limit	0.927	0.043	0.958	0.043

Table 3: Detection effectiveness for the fixed temperature threshold approach (baseline).

4. Discussion

The implemented image analysis pipeline that processes full 1280×1024 frames attains performance suitable for real-time processing. The estimated total system delays of 50 ms for the sequential CPU and 20 ms for the parallel GPU implementations are remarkably less than the established 110 ms. As a consequence, it would allow for operating closer to the temperature limits as reaction time is reduced. However, due to safety concerns, the limits will not be changed for the upcoming OP2.1. Furthermore, all processing steps in the sequential CPU implementation finish within 10 ms. Although the morphological reconstruction requires over 12 ms per image, the selected FH algorithm consists of three separable stages that can be pipelined provided the described pre-processing, which bounds the complexity of the morphological reconstruction.

The detection effectiveness evaluation shows that sensitivity equals 0.970 for 12074 archived valid discharge sequences. The high sensitivity corresponds to a situation in which PFCs were not damaged because the alarm was triggered in time before the overload. However, there were still 18 late alarms (FNs) that might threaten the integrity of the device. Justifications of late alarms are depicted in Appendix B in Table B.4 with corresponding cases listed in B.5. It is noteworthy that when the parallel GPU implementation is used, the number of late alarms is reduced to two since all false alarms with anticipation time longer or equal to -20 ms are detected in time. Moreover, it improves sensitivity (0.997) and lowers FPR (0.042).

The causes of FPs could be analysed because the protection system in OP1.2 was not connected to the interlock. Therefore, discharges were not interrupted when the overload occurred, providing an insight into what would have happened if the alarm had not been triggered. The relatively high FPR of 0.052, i.e., a discharge was interrupted needlessly, is often due to the temperature dropping soon after the alarm (see Figure 11) or due to operating the device close to its temperature limits (see Figure 12), where small rapid temperature fluctuations might trigger the alarm to prevent the overload. In both cases, this is the expected action of a protection system. The backward analysis underestimates TOD effectiveness since other external factors influencing the discharge are not considered, e.g., the heating is turned off. However, it presents a valuable baseline estimation of expected effectiveness in the upcoming OP2.1.

The majority of FPs occurred when the corrected surface temperature was close to 90% of the limit in the dynamic threshold approach (see Figure 13). For comparison, the sensitivity decreased by 0.007, and FPR increased by 0.043 in the corresponding fixed 90%

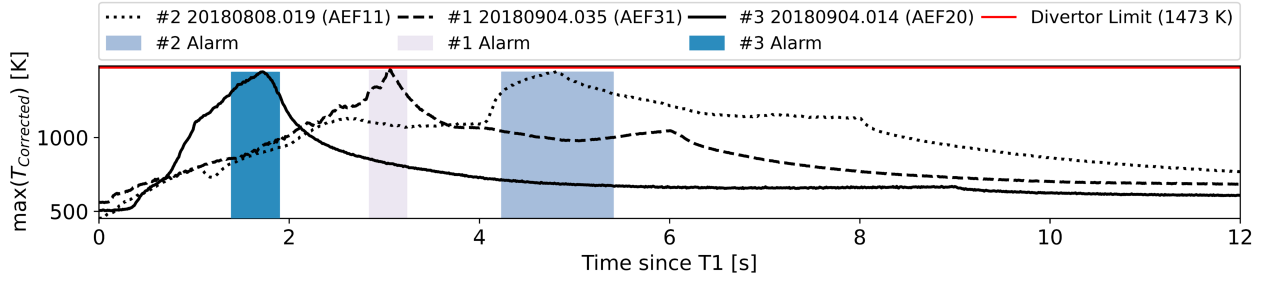


Figure 11: The first 12s of the three discharges where the heating was turned off after the alarm (False Positive (FP)). It shows the evolution of the corrected maximum surface temperature ($T_{corrected}$) on the divertors. The temperature rapidly grows, and TOD anticipates that the limit will be exceeded, but the heating is disabled before the limit is reached. Nevertheless, this is classified as FP in the statistical results.

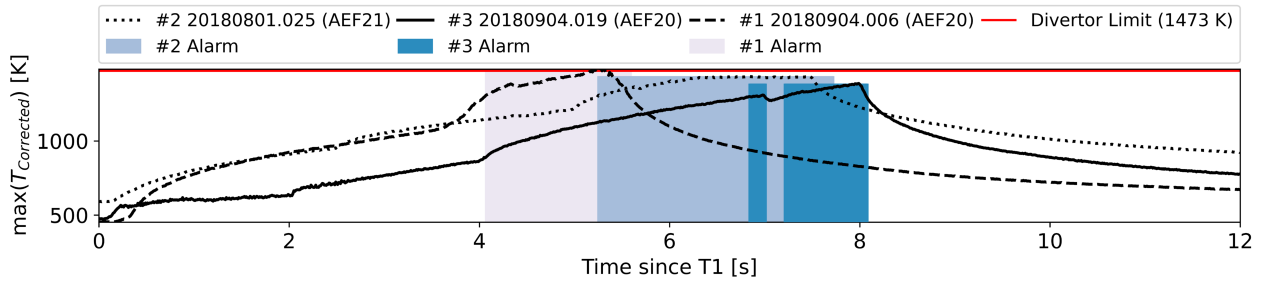


Figure 12: The first 12s of the three discharges where the device is operated close to its temperature limit (FP). It shows the evolution of the corrected maximum surface temperature ($T_{corrected}$) on the divertors. TOD anticipates that the limit will be exceeded since the temperature oscillates close to the limit, but it is not reached eventually. Nevertheless, this is classified as FP in the statistical results.

threshold approach. Therefore, with the justified FNs, the reduction of FPR is a priority. The dynamic approach attains lower FPR without compromising sensitivity. However, the fixed threshold approach benefits from the reduced processing delay compared to the slower CPU implementation of the dynamic threshold approach. The reduced processing delay increases sensitivity when the fixed threshold is set below 90%, in turn significantly increasing FPR. This disadvantage is resolved with the GPU implementation of the dynamic threshold approach, which equalises the delays between the approaches. In effect, the dynamic threshold approach increases the sensitivity and decreases FPR, improving detection effectiveness compared to the fixed threshold approach when its delay is reduced with parallel processing. The fixed threshold approach does not adapt to the rate of change. Consequently, either the threshold is low, resulting in high FPR or high, resulting in low sensitivity.

TOD uses the apparent heat flux, ignoring that there will be water cooling at the bottom during OP2.1. The heat flux approximation uses a semi-infinite solid model, i.e., without the cooled bottom. It was an accurate approximation for OP1.2 when PFCs were uncooled, even though some heat was dissipated due to radiation and heat conduction. This approximation is also sufficient for protection during OP2.1 with water-cooled PFCs because overloads can only occur during transient states when the heat flux changes, not in a steady-state when the temperature is constant. The model starts to deviate when the temperature approaches

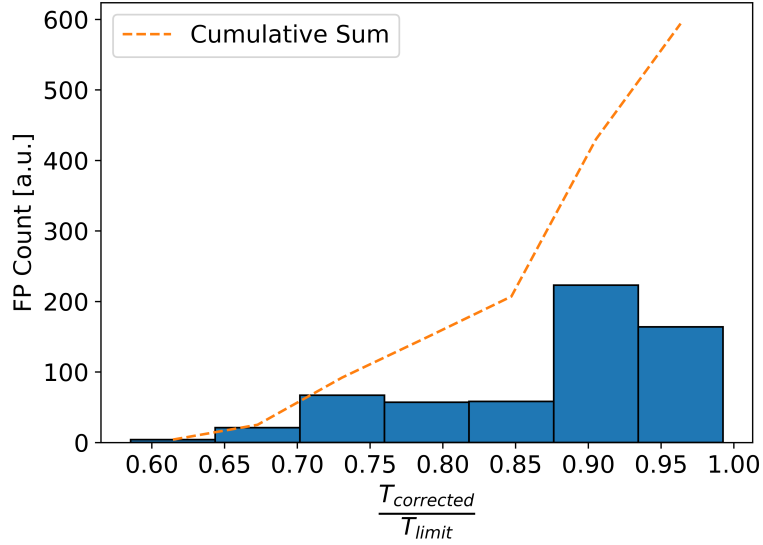


Figure 13: Distribution of FP cases for the dynamic threshold approach in terms of the ratio between the corrected surface temperature and limit. Most of the FPs occurred when the corrected surface temperature was close to 90% of the limit.

the final steady-state temperature. Therefore, this heat flux calculation is not valid for physics exploitation. The offline THERMAL Energy Onto DivertOR (THEODOR) code is an option to calculate the true heat flux for physics analysis [28]. In a steady-state, we observe a zero apparent heat flux as all the heat is removed from the bottom, and the temperature no longer increases. In this case, the dynamically calculated temperature threshold is equal to the temperature limit, and TOD never triggers unless there is already an overload. Such apparent high heat fluxes as on uncooled PFCs will not occur on water-cooled PFCs because the apparent heat flux approximately corresponds to the difference between the incoming and outgoing heat flux. Therefore, TOD will observe lower apparent heat fluxes for the same true incoming heat flux. As a consequence, measured changes between frames will be slower, providing more time to anticipate overloads within the delay time and reducing the magnitude of temperature fluctuations.

Moreover, observed leading edges [29], which caused some FNs, are not an imminent threat to tiles. The temperature limit is determined by the underlying cooling structures, which can be damaged and cause a water leak. A leading edge does not affect the bulk of the tile and hence the cooling structures. A leading edge typically self-heals because it erodes the material and finally disappears. The danger is that the eroded carbon can contaminate the plasma and cause a radiation collapse. According to the experiments conducted at W7-X during OP1.2 [29], the plasma was resilient to this contamination and did not collapse due to high radiation.

5. Conclusions

Machine protection systems are essential in all thermonuclear devices operating with high-power discharges. The implemented and evaluated TOD system for machine protection

fulfils the real-time constraints of 110 ms of total system delay and 10 ms per process. Moreover, it provides high detection sensitivity of 0.97 and 0.99 for all valid archived discharge sequences from the previous OP1.2 with the CPU and GPU implementation, respectively. The relatively high FPR is expected not to occur during the upcoming OP2.1 as the PFCs will be water-cooled, resulting in lower apparent heat flux. The attained acceleration is significant, i.e., a 95% and 99% decrease in runtime for the sequential CPU and parallel GPU implementations, respectively, compared to the Python prototype. As a consequence, more rapidly developing overloads can be detected in time. For the first time, the presented results confirm the feasibility of applying a full frame processing approach instead of an ROI approach to protect the W7-X stellarator in real-time by dynamically adjusting the temperature threshold. TOD comprises fundamentals for further advanced protection and control of thermal loads at W7-X, ultimately also at ITER [10].

The effectiveness of TOD also will be experimentally verified during the commissioning phase before OP2.1, and further adjustments of the algorithm parameters to the new OP2.1 acquisition system and FoVs are foreseen. Furthermore, AI solutions are to be researched to supplement TOD with intelligent scene understanding, e.g., to discriminate between true overloads and thermal events that affect the monitored surface temperature, such as reflections, leading edges or surface layers.

6. Acknowledgements

This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200—EUROfusion), Engineering Grants EEG21-17 and as part of co-financed by the Polish Ministry of Science and Higher Education within the framework of the international project International Projects Co-financed called 'PMW'. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

This scientific paper has been completed while the first author was the Doctoral Candidate in the Interdisciplinary Doctoral School at the Lodz University of Technology, Poland.

Appendix A. CUDA Queue of Pixels Reconstruction

The morphological reconstruction operates on two equally sized images, i.e., marker and mask, whose sizes are defined as WIDTH×HEIGHT. In the reconstruction by erosion, the following condition has to be fulfilled $\forall p \in D_{mask}, marker(p) \geq mask(p)$, where D_{mask} is a domain for both the marker and mask.

```
// Queue initialisation
ReadQueue = {}, lengthReadQueue = 0

for each thread with index (t):
    if  $t \in D_{mask}$  and
```



```

     $\exists q \in Neighbours_s(t)$  marker[q] > mask[q] and marker[q] > marker[p]:
    previousIndex = atomicAdd(&lengthReadQueue, 1)
    ReadQueue[previousIndex] = t

// Wavefront propagation
N = 8
NEIGHBOUR_OFFSETS[N] = { -WIDTH - 1, -WIDTH, -WIDTH + 1, // LT, Top, RT
                        -1, 1, // Left, Right
                        WIDTH - 1, WIDTH, WIDTH + 1 } // LB, Bottom, RB

WriteQueue = {}, lengthWriteQueue = 0

while lengthReadQueue > 0:
    for each thread with index (t):
        if t < N * lengthReadQueue:
            p = ReadQueue[t / N]
            q = p + NEIGHBOUR_OFFSETS[t % N]

            // Check the propagation condition
            if  $q \in Neighbours_s(p)$  and
                marker[q] != mask[q] and marker[q] > marker[p]:
                    previousValue = atomicMin(&marker[q], max(marker[p], mask[q]))

                    if previousValue > max(marker[p], mask[q]):
                        // Append to the queue
                        previousIndex = atomicAdd(&lengthWriteQueue, 1)
                        WriteQueue[previousIndex] = q

            // Each thread has its local pointer to both queues
            swapThreadLocalPointers(ReadQueue, WriteQueue)

        synchronizeAllGridThreads()

    if t == 0:
        // Update the lengths globally
        lengthReadQueue = lengthWriteQueue
        lengthWriteQueue = 0

    synchronizeAllGridThreads()

```

Listing 1: Compute Unified Device Architecture (CUDA) queue of pixels reconstruction by erosion

Kernels use grid-stride loops so that kernels do not have to be executed with a specific block and thread configuration to compute the correct result. However, the configuration determines the attained performance. All threads within the grid are synchronised with the cooperative groups introduced in CUDA 9.

Appendix B. Analysis of Late Alarms

Reason	Description
A	The strike-line extends towards the baffle next to the vertical target. There is a high heat flux at the edge of some tiles, and the temperature incenses too fast.
B	The manipulator plunging the plasma is visible. High heat fluxes appear in front of the stainless steel panels, which have a low temperature limit, triggering a false alarm.
C	The strike-line on the high-iota target extends towards the heat shield with high heat flux. The alarm is triggered but too late.
D	A hot spot develops on the baffle. A Neutral Beam Injection (NBI) blip [30] suddenly increases a heat flux and triggers the alarm. However, the temperature increase is too fast to react on time.
E	The Electron Cyclotron Resonance Heating (ECHR) fails, and the power hits the wall heat shields with a high heat flux triggering the alarm but too late.

Table B.4: Justifications for late alarms.

Program ID	Port	Anticipation Time [ms]	Overload Time since T1 [ms]	Reason
20171122.030	AEF51	-40	150	A
20171122.037	AEF51	-20	610	A
20180724.033	AEF30	-30	1570	B
20180821.023	AEF30	-10	6660	B
20180904.034	AEF20	-30	1960	C
20180911.035	AEF30	-40	3260	C
20180927.008	AEF41	-40	7060	C
20181016.040	AEF11	-40	1770	D
20181017.004	AEF51	-10	40	E
20181017.005	AEF51	-40	80	E
20181017.006	AEF51	-30	70	E
20181017.007	AEF51	-30	70	E
20181017.008	AEF51	-30	70	E
20181017.009	AEF51	-40	80	E
20181017.010	AEF51	-20	60	E
20181017.011	AEF51	-30	70	E
20181017.012	AEF51	-20	60	E
20181017.013	AEF30	-30	7490	C

Table B.5: Cases of late alarms when the anticipation time is shorter than -50 ms. T1 is the timestamp of the first frame of the sequence.

Figures B.14, B.15, B.16, B.17 and B.18 illustrate examples for each late alarm justification. The images show the maximum corrected surface temperature superimposed on the W7-X Computer-Aided Design (CAD) models.

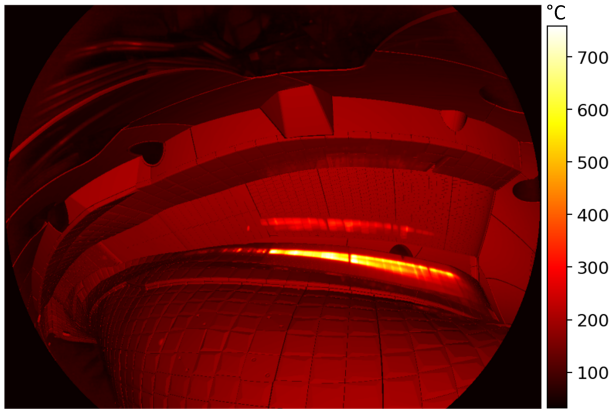


Figure B.14: A (20171122.030 AEF51).

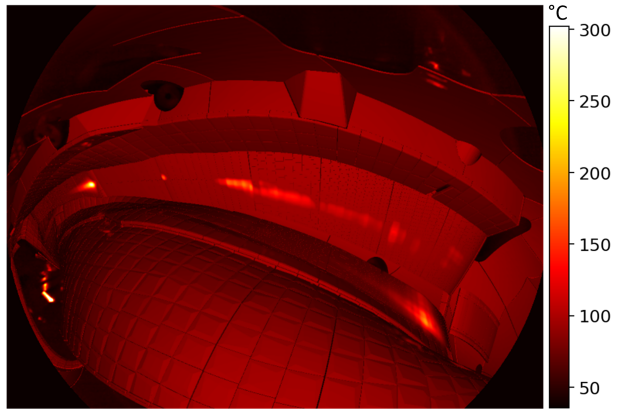


Figure B.15: B (20180724.033 AEF30).

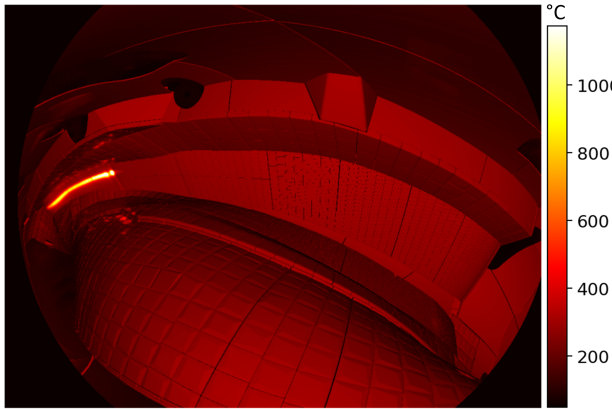


Figure B.16: C (20180904.034 AEF20).

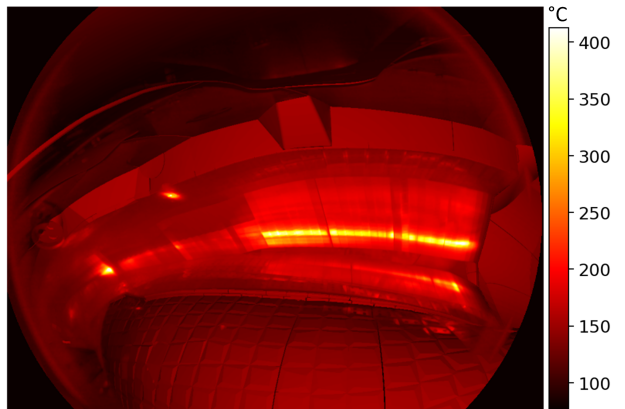


Figure B.17: D (20181016.040 AEF11).

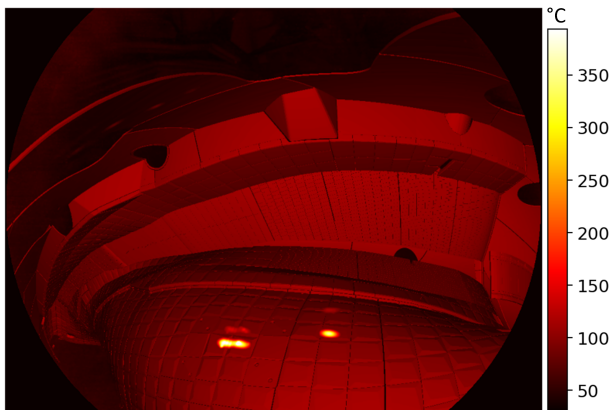


Figure B.18: E (20181017.005 AEF51).

References

- [1] A. Puig Sitjes, B. Jabłoński, Y. Gao, V. Moncada, F. Pisano, S. Fischer, M. Jakubowski, D. Makowski, D. Hathiramani, T. Sunn Pedersen, A. Winter, Thermal loads protection of Wendelstein 7-X divertors with real-time infrared imaging, poster at 32nd Symposium on Fusion Technology, Dubrovnik, Croatia (2022).
- [2] M. Jakubowski, P. Drewelow, J. Fellingner, A. Puig Sitjes, G. Wurden, A. Ali, C. Biedermann, B. Cannas, D. Chauvin, M. Gamradt, H. Greve, Y. Gao, D. Hathiramani, R. König, A. Lorenz, V. Moncada, H. Niemann, T. T. Ngo, F. Pisano, T. Sunn Pedersen, Infrared imaging systems for wall protection in the W7-X stellarator (invited), *Review of Scientific Instruments* 89 (10) (2018) 10E116. doi:10.1063/1.5038634.
- [3] J. Fellingner, U. Lippmann, H. Greve, M. Alhashimi, M. Schülke, S. Äkaslompolo, P. Drewelow, M. Jakubowski, R. König, A. Lorenz, Design of endoscopes for monitoring water-cooled divertor in W7-X, *Fusion Engineering and Design* 158 (2020) 111841. doi:10.1016/j.fusengdes.2020.111841.
- [4] J. Fellingner, M. Alhashimi, M. Gamradt, Y. Gao, H. Greve, F. Herold, M. Jakubowski, U. Lippmann, A. Lorenz, A. Ott, K. Risse, D. Rondeshagen, M. Schülke, E. Sözer, Manufacturing of endoscopes for monitoring water-cooled divertor in Wendelstein 7-X, poster at 4th European Conference on Plasma Diagnostics, Virtual (2021).
- [5] R. Mitteau, C. Belafdil, C. Balorin, X. Courtois, V. Moncada, R. Nouailletas, B. Santraine, WEST operation with real time feed back control based on wall component temperature toward machine protection in a steady state tungsten environment, *Fusion Engineering and Design* 165 (2021) 112223. doi:10.1016/j.fusengdes.2020.112223.
- [6] V. Huber, A. Huber, D. Kinna, G. Matthews, G. Sergienko, I. Balboa, S. Brezinsek, P. Lomas, J. Mailoux, P. McCullen, P. Mertens, F. Rimini, S. Silburn, D. Valcarcel, K.-D. Zastrow, The software and hardware architecture of the real-time protection of in-vessel components in JET-ILW, *Nuclear Fusion* 59 (7) (2019) 076016. doi:10.1088/1741-4326/ab1a79.
- [7] R. Drube, G. Neu, R. H. Cole, K. Lüddecke, T. Lunt, A. Herrmann, The ASDEX Upgrade digital video processing system for real-time machine protection, *Fusion Engineering and Design* 88 (11) (2013) 2870–2874. doi:10.1016/j.fusengdes.2013.05.065.
- [8] A. Herrmann, B. Sieglin, M. Faitsch, A. U. Team, Surface temperature measurement of in-vessel components and its real-time validation, *Fusion Science and Technology* 69 (3) (2016) 569–579. doi:10.13182/FST15-187.
- [9] A. Puig Sitjes, M. Jakubowski, D. Naujoks, Y. Gao, P. Drewelow, H. Niemann, J. Fellingner, V. Moncada, F. Pisano, C. Belafdil, R. Mitteau, M.-H. Aumeunier, B. Cannas, J. R. Casas, P. Salembier, R. Clemente, S. Fischer, A. Winter, H. Laqua, T. Bluhm, K. Brandt, T. W.-X. Team, Real-time detection of overloads on the plasma-facing components of Wendelstein 7-X, *Applied Sciences* 11 (24) (2021). doi:10.3390/app112411969.
- [10] R. Reichle, P. Andrew, G. Counsell, J.-M. Drevon, A. Encheva, G. Janeschitz, D. Johnson, Y. Kusama, B. Levesy, A. Martin, C. S. Pitcher, R. Pitts, D. Thomas, G. Vayakis, M. Walsh, Defining the infrared systems for ITER, *Review of Scientific Instruments* 81 (10) (2010) 10E135. doi:10.1063/1.3491199.
- [11] F. Orsitto, R. Villari, F. Moro, T. Todd, S. Lilley, I. Jenkins, R. Felton, W. Biel, A. Silva, M. Scholz, J. Rzakiewicz, I. Duran, M. Tardocchi, G. Gorini, C. Morlock, G. Federici, A. Litnovsky, Diagnostics and control for the steady state and pulsed tokamak DEMO, *Nuclear Fusion* 56 (2) (2016) 026009. doi:10.1088/0029-5515/56/2/026009.
- [12] J.-M. Travere, M.-H. Aumeunier, M. Joanny, T. Loarer, M. Firdaouss, E. Gauthier, V. Martin, V. Moncada, L. Marot, D. Chabaud, E. Humbert, J.-J. Fermé, C. Thellier, Imaging challenges for ITER plasma-facing component protection, *Fusion Science and Technology* 64 (4) (2013) 735–740. doi:10.13182/FST13-A24093.
- [13] M.-H. Aumeunier, M. Le Bohec, R. Brunet, A. Juven, M. Adel, X. Artusi, R. Miorelli, C. Reboud, F. Rigollet, Development of inverse methods for infrared thermography in fusion devices, *Nuclear Materials and Energy* 33 (2022) 101231. doi:10.1016/j.nme.2022.101231.
- [14] E. Grelier, R. Mitteau, V. Moncada, Deep learning and image processing for the automated analysis of

- thermal events on the first wall and divertor of fusion reactors, *Plasma Physics and Controlled Fusion* 64 (2022) 104010. doi:10.1088/1361-6587/ac9015.
- [15] B. Jabłoński, D. Makowski, P. Perek, Implementation of thermal event image processing algorithms on NVIDIA Tegra Jetson TX2 embedded system-on-a-chip, *Energies* 2021 (15) (14). doi:10.3390/en14154416.
- [16] B. Jabłoński, D. Makowski, P. Perek, P. Nowak vel Nowakowski, A. Puig Sitjes, M. Jakubowski, Y. Gao, A. Winter, T. W.-X. Team, Evaluation of NVIDIA Xavier NX platform for real-time image processing for plasma diagnostics, *Energies* 15 (6) (2022). doi:10.3390/en15062088.
- [17] R. Vilbrandt, J. Schacht, U. Herbst, S. Marsen, K. Rahbarnia, K. J. Brunner, M. Hirsch, J. Knauer, D. Naujoks, S. Degenkolbe, H.-S. Bosch, The fast interlock system of W7-X – first experience, *Fusion Engineering and Design* 151 (2020) 111380. doi:10.1016/j.fusengdes.2019.111380.
- [18] J. Schacht, A. Wölk, S. Degenkolbe, U. Herbst, S. Pingel, E. Scharff, Enhancements of the fast interlock system for Wendelstein 7-X operational phase OP2.1, poster at 23rd IEEE Real Time Conference, Virtual (2022).
- [19] R. C. Gonzalez, R. E. Woods, S. L. Eddins, *Digital image processing using MATLAB*, 3rd Edition, Gatesmark Publishing, 2020.
- [20] L. Vincent, Morphological grayscale reconstruction in image analysis: applications and efficient algorithms, *IEEE Trans Image Process* 2 (2) (1993) 176–201.
- [21] M. Roynard, E. Carlinet, T. Géraud, An image processing library in modern C++: Getting simplicity and efficiency with generic programming, in: *Reproducible Research in Pattern Recognition*, Springer International Publishing, Cham, 2019, pp. 121–137. doi:10.1007/978-3-030-23987-9_12.
- [22] K. Robinson, P. F. Whelan, Efficient morphological reconstruction: a downhill filter, *Pattern Recognition Letters* 25 (15) (2004) 1759–1767. doi:10.1016/j.patrec.2004.07.002.
- [23] G. Teodoro, T. Pan, T. M. Kurc, J. Kong, L. A. Cooper, J. H. Saltz, Efficient irregular wavefront propagation algorithms on hybrid CPU–GPU machines, *Parallel Computing* 39 (4) (2013) 189–211. doi:10.1016/j.parco.2013.03.001.
- [24] S. Hong, S. K. Kim, T. Oguntebi, K. Olukotun, Accelerating CUDA graph algorithms at maximum warp, *SIGPLAN Not.* 46 (8) (2011) 267–276. doi:10.1145/2038037.1941590.
- [25] J. Gil, R. Kimmel, Efficient dilation, erosion, opening, and closing algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (12) (2002) 1606–1617. doi:10.1109/TPAMI.2002.1114852.
- [26] H. Laqua, T. Bluhm, M. Grahl, M. Grün, C. Hennig, A. Holtz, J. Krom, G. Kühner, M. Lewerentz, H. Riemann, J. Schacht, A. Spring, A. Werner, Experiences with the segment control system at Wendelstein 7-X operation, *Fusion Engineering and Design* 123 (2017) 588–592. doi:10.1016/j.fusengdes.2017.03.066.
- [27] T. S. Pedersen, R. König, M. Krychowiak, M. Jakubowski, J. Baldzuhn, S. Bozhenkov, G. Fuchert, A. Langenberg, H. Niemann, D. Zhang, K. Rahbarnia, H.-S. Bosch, Y. Kazakov, S. Brezinsek, Y. Gao, N. P. and, First results from divertor operation in Wendelstein 7-X, *Plasma Physics and Controlled Fusion* 61 (1) (2018) 014035. doi:10.1088/1361-6587/aaec25.
- [28] A. Herrmann, W. Junker, K. Gunther, S. Bosch, M. Kaufmann, J. Neuhauser, G. Pautasso, T. Richter, R. Schneider, Energy flux to the ASDEX-upgrade diverter plates determined by thermography and calorimetry, *Plasma Physics and Controlled Fusion* 37 (1) (1995) 17–29. doi:10.1088/0741-3335/37/1/002.
- [29] A. P. Sitjes, Y. Gao, M. Jakubowski, P. Drewelow, H. Niemann, A. Ali, V. Moncada, F. Pisano, T. Ngo, B. Cannas, M. Slecicka, Observation of thermal events on the plasma facing components of Wendelstein 7-X, *Journal of Instrumentation* 14 (11) (2019) C11002–C11002. doi:10.1088/1748-0221/14/11/c11002.
- [30] Y. Kolesnichenko, V. Lutsenko, Populations of fast ions produced by neutral beam injection pulses, *Nuclear Fusion* 59 (12) (2019) 126005. doi:10.1088/1741-4326/ab3a57.