

UCL CDT DIS Note

UCLCDTDIS-2019

22nd February 2021



Calibration of Diagnostic Image Data in Fusion Experiments

Christian Gutschow^a, Augustin Marignier^a, Jamie McGowan^a, Emily Nurse^a, Scott Silburn^b, and Sam Van Stroud^a

^aUniversity College London

^bCulham Centre for Fusion Energy

Fusion energy is a promising solution to the problem of generating safe, clean energy at scale. The next generation of fusion reactors, loaded with advanced sensing and diagnostic subsystems, aim to demonstrate net energy gain. A key challenge remains the lack of a predictive understanding of the dynamics of the confined plasma in the reactors. Diagnostic data, such as images and videos of the reactors' interiors, are crucial for constraining models of the plasma and improving the understanding of how the machine is operating. However, maximising the potential of the camera systems in particular is hampered by challenges related to the harsh operating conditions inside the reactors. Recorded images and videos are of low visual quality, and movement of the cameras and optics systems makes analysing the plasma with respect to a fixed coordinate system challenging. This project aims to scope solutions to these problems using state of the art image processing and stabilisation techniques. Our results contain promising solutions that alleviate the worst of these challenges.

Contents

1	Introduction	1
1.1	Background	1
1.2	Problem Statement	1
1.3	Data	2
2	Image Preprocessing	3
2.1	Image Metrics	3
2.2	Clustering	4
2.3	Sequential	4
3	Image Stabilisation	4
3.1	Feature Matching	5
3.1.1	Line Segment Detection	5
3.1.2	DeepMatching	6
3.2	Image Transformation	6
3.3	Scoring Aligned Images	7
4	Results	8
5	Conclusions	9
A	Methods	11
A.1	Preprocessing Operations	11
A.1.1	Gaussian Subsampling	11
A.1.2	Histogram Equalisation	11
A.1.3	Denoising	12
A.2	Effect on Image Metrics	12
A.3	Image Transformations	12
A.4	Scoring	14
A.4.1	True Alignment Detection	14
A.4.2	Reducing the Impact of Noise	14
A.4.3	Subblock Scoring	15
A.4.4	Automatic Tolerance Detection	15
A.4.5	Automatic Image Cropping	15
B	Results	16
B.1	MAST Video Stabilisation	16

1 Introduction

Nuclear fusion is the process that powers the Sun, and all other stars. It is therefore one of the most important and powerful processes in the universe. When light nuclei fuse to form a larger nucleus, the sum of the masses of light nuclei is greater than the mass of the new larger nucleus. Thanks to energy-mass equivalence, this mass difference is released as excess energy, thus forming the basis of nuclear fusion power. All life on Earth would not be possible without nuclear fusion and soon all life on Earth might owe even more to this incredible mechanism.

1.1 Background

Peaceful means of harnessing nuclear energy have been researched into by the UK Atomic Energy Authority (UKAEA) since 1954. More recently, the focus has been on toroidal nuclear fusion tokamak reactors, designed in the 1960s in the Soviet Union and the most developed shape for a nuclear fusion reactor. They use toroidal and poloidal magnetic fields to confine plasma within their walls. The aim of these reactors is to research how nuclear fusion scales up to the size of a fully operational nuclear reactor that can feed power onto a country's national grid. The Joint European Torus (JET) and Mega Amp Spherical Tokamak (MAST) are two machines housed at the Culham Centre for Fusion Energy (CCFE). JET is the flagship device for EUROfusion, the EU wide programme for developing fusion power [1]. It is also currently the largest operating tokamak in the world, the only one capable of running Deuterium-Tritium mixtures, and as of 1997 holds the world record for fusion power gain. MAST-Upgrade, set to come online later in 2019, will also participate in the EUROfusion programme.

The plan for nuclear fusion reactors is to heat two different isotopes of Hydrogen to extremely high temperatures, via a number of different heating mechanisms. Deuterium (^2H) and Tritium (^3H) are fused together to form Helium and a neutron, this neutron is then slowed down in a dense medium which collects its energy. Deuterium can be extracted from water and Tritium can be created from Lithium¹ in the Earth's crust, this means that the supplies of fuel will last for millions of years.

Nuclear fusion has many advantages compared to other types of power stations in use at the moment. First and foremost is the carbon-free emissions, as the only 'waste products' from a nuclear fusion reactor are small amounts of Helium which will not contribute to atmospheric pollution. There is also the advantage of not having to deal with any long lived radioactive waste, the massively increased energy efficiency², the increased safety and reliability and many more.

1.2 Problem Statement

Cameras are vital tools for understanding the processes inside fusion reactors. They can be sensitive to a wider range of the electromagnetic spectrum than the human visual system, and dedicated high speed cameras can record transient events at high frame rates, rendering them amenable for analysis. There are three general categories for the cameras. Firstly, visible light cameras are used for operators to visually inspect what is happening inside the machine. Secondly, cameras are used for shape control and machine protection. For example, high-speed visible light cameras with real-time processing determine the shape of the plasma and feedback directly to the shape controller. Additionally, disruption events, which are a sudden loss of plasma confinement, happen on timescales of tens of millisecond and can result in serious and costly damage to the reactor, hence the need to monitor them with high-speed cameras. The third category of cameras is the science cameras. These can be sensitive to various different wavelengths and very high frame rate, used for observing, for example, atom and ion spectral lines, dynamics of plasma turbulence or

¹ Due to Tritium's short half-life (12.5 years), it is not naturally occurring. Ideally, fusion reactors will become self-sufficient in Tritium by using neutrons from fusion to fission Lithium-6

² One kilogram of fuel can supply as much energy as 10 million kilograms of fossil fuel.

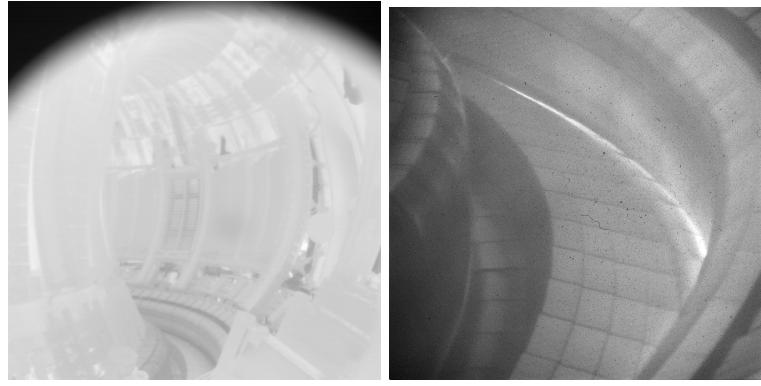


Figure 1: Two camera views into JET.

calculating heat flux on the walls of the tokamak. All this can aid in the development of plasma modelling and greater theoretical understanding of the plasma required for the growth of nuclear fusion power.

Knowing the spatial calibration and viewing geometry of the cameras is important for enabling the quantitative analysis of the image data. However, the cameras in a tokamak are not entirely stable. The tokamak itself shakes during operation, causing the cameras to move resulting in images that do not show exactly the same view. This project investigates how to calibrate images such that they show the same view to facilitate comparison and analysis, implementing this for two different scenarios. Firstly, correcting for long term camera drift between images collected at two arbitrarily different times. Secondly, correcting for frame-to-frame motion within a video.

Currently, calibration is done manually using a point-picking program, CCFE, developed at UKAEA. This project aims to automate this process, making it as general as possible so that the corrections work on a wide range of image qualities and on different camera views.

1.3 Data

The data are sets of still images and videos from visible light and infrared cameras looking into the JET (Figure 1) and MAST (Figure 11) tokamaks. JET produces on the order of 20 GB for every ~30 s pulse video³, with pulses occurring up to every 20 minutes. MAST-Upgrade is expected to produce over 10 GB per pulse, even for pulses shorter than a second, due to the extremely high camera frame rates. All this data further shows the need for an automated calibration process.

Our still images are disruption images. A disruption is a sudden loss of the plasma confinement which ends the plasma pulse, caused from any plasma instabilities or various technical issues. Disruptions are so violent and short-lived that they can be seen in only one or two frames of a pulse video. As such, all pairs of images (e.g. Figure 2) are separated by an arbitrary amount of time. For a lot of the camera systems on JET, disruptions are the only source of light bright enough to get clear images of the vessel structure for doing the camera alignment. Experimental campaigns in JET involve plasmas being created and because of this, at the start of experimental campaigns, a number of disruptions are deliberately caused, specifically for camera calibration purposes.

Figure 2 shows a sample pair of disruption images in JET in grayscale, and a zoomed overlay of the two images in false colour. Immediately noticeable are differences in image exposure and contrast, and the distributions of plasma in the images. From the coloured image it is clear that the two are not perfectly aligned, as indicated by the subhorizontal double lines of the tiled area. It is this misalignment that this work seeks to correct.

³ When JET first came online in 1984 it produced only ~4 MB

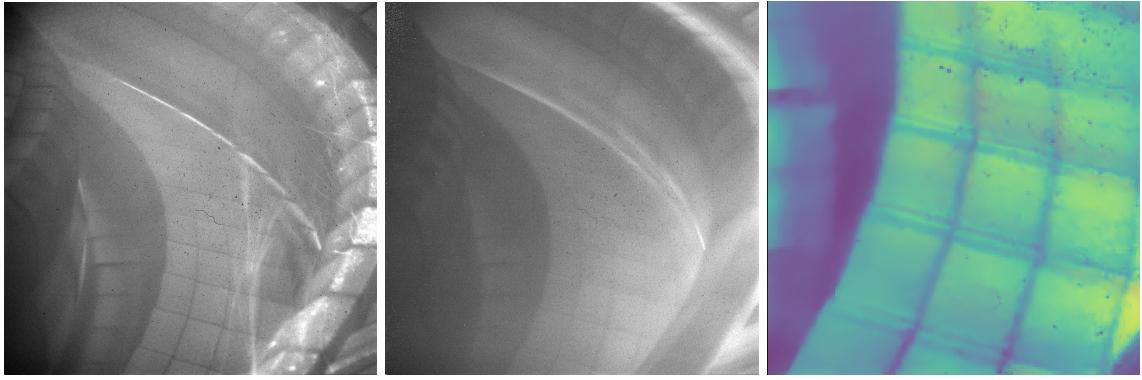


Figure 2: Greyscale: A pair of disruption images form JET, as seen from camera KL11. Colour: A zoom on the tiled section of the view from camera KL11, where the two images have been overlaid, showing relative motion between the two frames.

2 Image Preprocessing

Preprocessing is an essential part of the process of image calibration. The data from the camera's in the experiments are all of varying quality and so for a computer to even be able to tell that two images are of the same view a level of preprocessing is required. This is even more important for this project as the images also need to be calibrated as accurately as possible. Image manipulations, including Gaussian downsampling, histogram equalisation and noise filtering, are implemented using the OPENCV package [2]. Detailed descriptions can be found in Appendix A.1. Preprocessing is performed to improve image quality and to reduce variation between images, so that feature matching algorithms are more easily able to find correspondences between images. However, this cannot be done manually for each image. This procedure is to be automated, operating under the assumption that similar images will respond in a similar way to the same preprocessing.

While the feasibility of both the clustering and sequential preprocessing paradigms have been demonstrated, the final choice of preprocessing will be based on a quantitative measure of which preprocessing method produces the best alignments (Section 4).

2.1 Image Metrics

To determine how similar images are and how they respond to preprocessing, six different metrics are calculated for each image: BRISQUE [3], noise, global contrast, and local contrast metrics. These are calculated by splitting the image into subblocks and calculating for each image: the mean subblock contrast, the standard deviation of subblock mean, and the standard deviation of the subblock contrast. BRISQUE gives a holistic no-reference measure of quality of the image based on statistics of luminance coefficients [3]. Noise is measured as the variance of the Laplacian (VoL) of the image. The Laplacian of the image can be computed from the convolution of a Laplacian kernel (representing second order spatial derivatives) with the pixel intensities of the image. The variance of the resulting edge map is then calculated. A small VoL indicates a blurred image and a large VoL indicates a un-blurred image. In practice, more direct noise metrics (e.g. the mean squared error in pixel intensity between the original image and a smoothed copy) were found to be very tightly correlated with the VoL blur metric, and so VoL blur was used as a proxy for noise. Contrast is measured as the standard deviation of the global histogram of pixel intensities of the image. The last three metrics are local metrics, in that they are based on the histograms of subblocks of the image. They describe the local contrasts and how these change throughout the image.

2.2 Clustering

On the assumption that similar images react in similar ways to the same preprocessing, different ways of clustering the images are investigated, then determining the optimal preprocessing on each cluster. Looking in the 6D metric space, attempts of clustering were made based on k -nearest neighbours, nearest neighbours by radius and dividing the images into a fixed number of clusters. These all lead to promising results in the sense that images in a cluster look more similar by eye after preprocessing than before, however the effect this has on feature matching (discussed in Section 3) and calibration is difficult to quantify. This makes it difficult to determine the optimum preprocessing of each cluster.

The fixed number of clusters mechanism with varying cluster sizes seemed to produce the clearest clusters to the eye. For example, for a set of 109 images from camera KL11 (e.g. Figure 2) 6 clusters were formed. Clustering was performed using the image metrics discussed in Section 2.1. This was promising and when different preprocessing parameters were applied to each cluster, the result produced images that, from observation, were even more similar to each other.

2.3 Sequential

An alternative to preprocessing images based on their clustering is to sequentially apply different preprocessing operations to each image. In this paradigm, a single preprocessing operation is performed at a time, and the parameters for this operation are chosen based on the metrics calculated from the current state of the image. After this single preprocessing operation has been applied, the image metrics are recalculated, and the next preprocessing operation is performed on the updated image, using the updated image metrics. The order of preprocessing operations, determined through extensive testing, was as follows: Gaussian subsampling, local histogram equalisation, bilateral filtering, non-local means filtering (Appendix A.1). Of these, the local histogram equalisation and non-local means filtering operations relied on parameters that were dependent on the image metrics. The Gaussian subsampling and bilateral filtering steps are applied with the same parameters to all images.

In order to approximate the best choice of parameters to use for every image, manual optimisation of the preprocessing parameters for each operation was performed for a sample of images that is representative of the varying quality, exposures and plasma contents of the full image data set. A numerical function was fit to this sample, where the x values were the values of a given metric (for example the noise metric for the non-local means), and the y values are the values of the preprocessing parameter that is applied to that image (for example the strength of the non-local means filter). The numerical fit allows for interpolation of the sampled points so that any image can be preprocessed based on its metrics. The clip limit of the local histogram equalisation algorithm was fit using an exponential function, while the strength of the non-local means filter was fit with a linear function. In order to reduce the complexity of this optimisation process, the other parameters of these operations were kept constant.

Applying preprocessing sequentially allows preprocessing operations to be targeted at a specific feature of the image, for example the level of noise. As histogram equalisation increases the amount of noise in an image, it makes sense to determine how much denoising to perform (by calculating the relevant noise metric) only after having applied histogram equalisation. The effects of applying sequential preprocessing on the distribution of metrics can be found in Appendix A.2.

3 Image Stabilisation

The process of aligning an image l_1 with respect to some other image l_2 can be factorised into two distinct steps. Firstly, common features in the two images must be detected, and a per feature correspondence must be known. The relative positions of the corresponding features in the

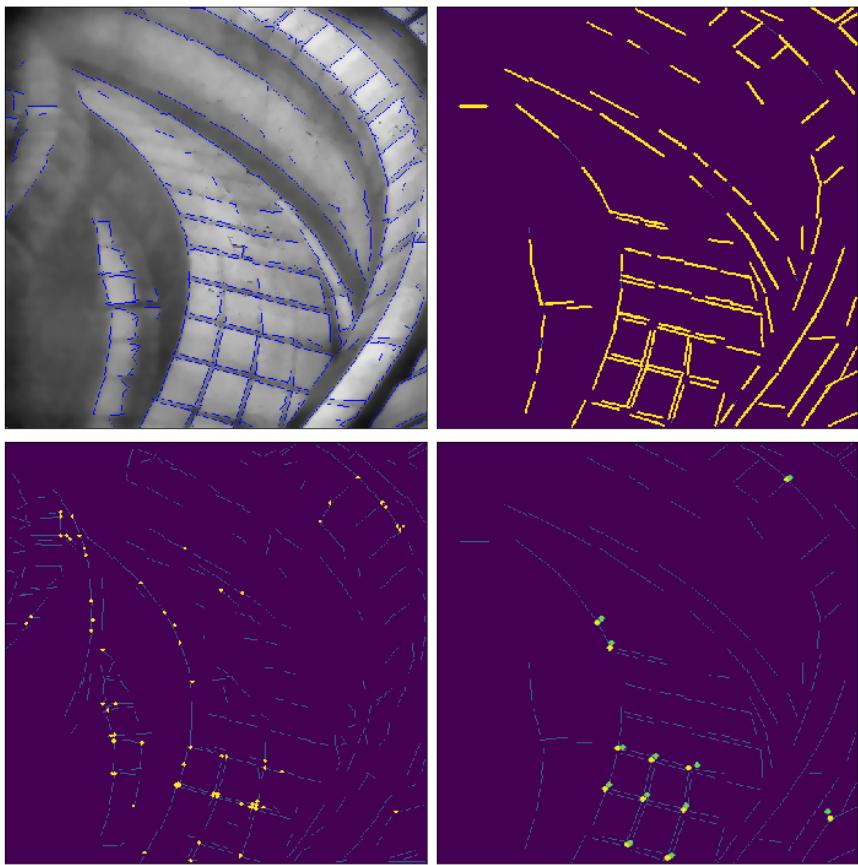


Figure 3: (top left) LSD results on a preprocessed KL11 image. Line segments detected by the LSD are shown in blue. (top right) Result of filtering and clustering line segments. The resulting processed line segments can be extended in either direction, allowing for (bottom left) intersections to be detected. (bottom right) Common point pairs between a pair of images. The green circles show the detected points for one image, and the corresponding points in another image are shown in yellow.

images can then be used to determine how the camera has moved between the shots. Finally, a transformation can be applied to l_1 that corrects for its relative movement to l_2 .

3.1 Feature Matching

Different methods to extract and correlate features in similar images are trialed. The main goal here is to identify points that are common between images which can be used to determine the transformation.

3.1.1 Line Segment Detection

A first attempt at feature matching used the Line Segments Detector (LSD) available in OPENCV [2]. The algorithm used is based on growing regions of pixels of similar gradient. Full details of the algorithm, including its validation of line segments, can be found in [4]. One issue with this method is that it produces straight line segments (left hand plot of Figure 3) rather than individual points needed to accurately determine transformations between images. Additionally, this approximates curved features as multiple end-to-end straight segments. As can also be seen in the left hand plot of Figure 3, noise is detected by the LSD. These noise segments are removed by filtering out segments shorter than a certain length. The remaining segments are extended to find intersections (right hand plot of Figure 3), which give the points to be used to determine the transformation.

Performing this for a pair of images finds corresponding point pairs from which the transformation can be calculated (Figure 3). The main issue with this is the small number of point pairs and their poor coverage of the image. In areas where many point pairs are found the transformation

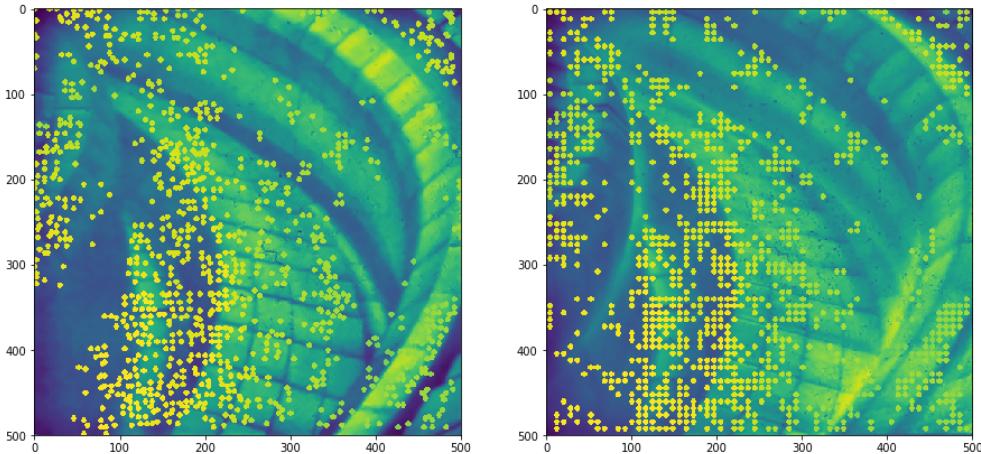


Figure 4: Point pairs between a pair of images using DeepMatching. Points are coloured according to the internal score of the match (yellow is higher). As opposed to the sparse feature detection in Figure 3, DeepMatching generates a dense correspondence between points in two images.

works relatively well. However, there were a few point pairs where the homography (a relationship in space) is unconstrained and the transformation fails. Initial attempts at finding a homography between two images based on manually selected point pairs in the lower region of the KL11 data proved unsuccessful. The homography was unconstrained in areas of the image where no point pairs were present. Some progress was made in generating point pairs across all regions of the image, but it was not possible to guarantee sufficient point pairs to produce a successful and constrained homography in every case. For this reason, a more advanced approach for feature matching was used, discussed in Section 3.1.2.

3.1.2 DeepMatching

As an alternative to LSD, the state-of-the-art dense feature matching algorithm DeepMatching [5] was investigated. DeepMatching performs well for image pairs with non-rigid transformations (i.e. transformations that are not simply 2D translations) and repetitive textures, the latter being particularly relevant for the images in question here. The architecture of DeepMatching is loosely inspired by deep convolutional networks. Figure 4 shows clearly the high density of corresponding points in comparison to the LSD approach in Figure 3. However, as individual point pairs may be of lower quality, and a homography only needs four point pairs to be fully constrained [2], a method of generating homographies from repeated random sampling of point pairs was employed [6]. Generating homographies in this way significantly improved the image alignment.

3.2 Image Transformation

Once corresponding features from a pair of images have been obtained, they can be used to recover the transformation that relates the images. Features in the images are assumed to behave as rigid bodies. It is additionally assumed that the image is of a planar surface (i.e. contains no depth). These assumptions allow the use of a homography model to relate the two images. While both of these assumptions are broken for the images under consideration (the plasma in the images does not behave as a rigid body, and the images have depth), analysis of the results suggest that there is enough freedom in the homography transformation to obtain a successful match. As such, the idealised homography transformation model is an acceptable approximation of the true transformation between images. It is likely that the increased complexity introduced by a more general transformation model would outweigh the extra freedom provided in these transformations. However, alternative transformation models not relying on the assumptions of homographies are detailed in Appendix A.3.

3.3 Scoring Aligned Images

A method for scoring how well images have been aligned is desirable for two reasons. Firstly, it allows one to optimise the preprocessing of images, and to make comparisons between different feature matching algorithms. Secondly, matches that score poorly can be discarded at the final stage, and therefore prevented from being used in any subsequent physics analysis. A custom scoring algorithm, DDScore, has been developed for this purpose. The core algorithm is discussed in this section, while extensions and modifications to the algorithm are reserved for Appendix A.4.

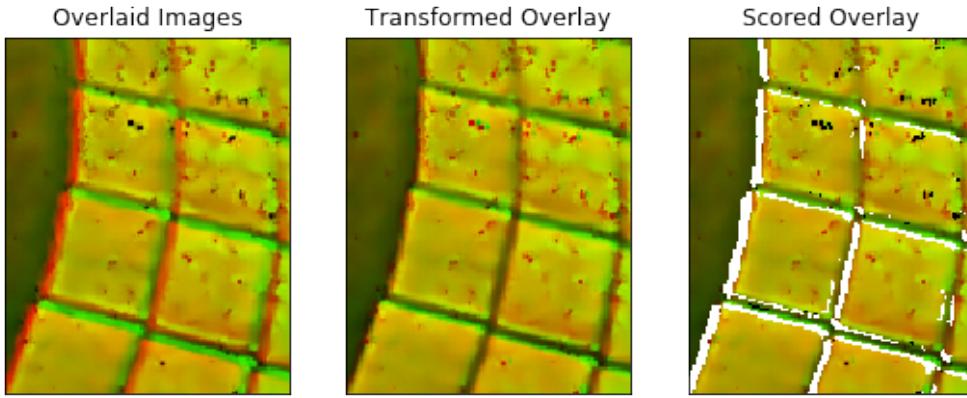


Figure 5: (left) Initial grayscale images l_1 (in the green channel) and l_2 (in the red channel) overlaid, showing red and green fringes at positions where there is a large difference between intensities of the two images. (middle) An overlay of l'_1 and l_2 , where l'_1 has been created using a homography generated by point pairs from DeepMatching. (right) The positions in the image that are scored as improved are coloured in white, while those detected as having deteriorated are coloured in black.

As inputs DDScore takes two images l_1 and l_2 that are to be matched, and a transformed version of l_1 , called l'_1 , that is generated from a matching between l_1 and l_2 . For each i, j pixel location in the image, d^{ij} is calculated as

$$d^{ij} = |l_1^{ij} - l_2^{ij}| - |l'_1{}^{ij} - l_2^{ij}|, \quad (1)$$

where l^{ij} denotes the pixel intensity of image l at the ij th pixel. d^{ij} quantifies how the difference in pixel intensity between $l_1^{(i)}$ and l_2 has changed over the course of the alignment. If, at a given pixel location, the images l_1 and l_2 are not aligned, then the absolute value of the difference $|l_1^{ij} - l_2^{ij}|$ will be large. If, after matching, the images are aligned, $|l'_1{}^{ij} - l_2^{ij}|$ will be small. Hence d^{ij} at this location will be positive. Conversely for points that were initially aligned and moved out of alignment, d^{ij} will be negative. In order to detect only points that have significantly changed, and not natural variation in intensity in the image, a ternary thresholding schema

$$s^{ij} = \begin{cases} 0 & \text{if } |d^{ij}| < t \\ 1 & \text{if } d^{ij} > t \\ -1 & \text{if } d^{ij} < -t \end{cases} \quad (2)$$

is used, where t is a tolerance parameter used to reject noise in the score that results from natural intensity variations in the images (this parameter is discussed in Appendix A.4.4). An example of the scoring algorithm is shown in Figure 5. Finally, given the values of s^{ij} across the image, the algorithm returns S , which is the fraction of changed points that have improved (i.e. the ratio of the number of points i, j where $s^{ij} = 1$ to the number of points where $s^{ij} \neq 0$), along with a count of the number of points that significantly changed (i.e. the number of points i, j where $s^{ij} \neq 0$).

The score depends on the image content and therefore also on any preprocessing that has been applied to the images. Thus, in order that the match quality can be compared across different

preprocessing strategies, initial images are fed to the scoring algorithm after passing through a fixed preprocessing step that is has been chosen to produce the most accurate score. This choice is discussed in Section 4. An overview of the flow of images through the framework is shown in Figure 6.

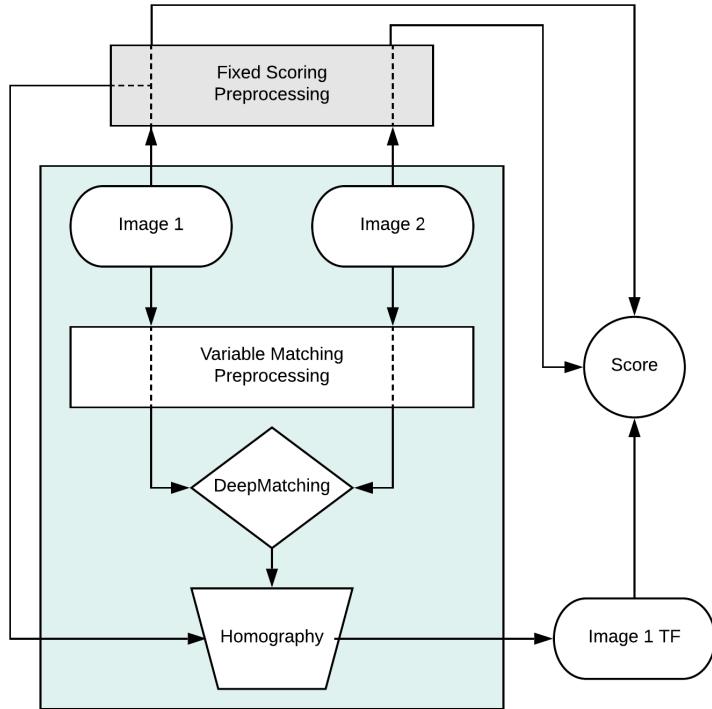


Figure 6: A high level overview of the alignment framework. In the green shaded area the alignment takes place, and the variable matching preprocessing is optimised by the scoring algorithm, which is shown outside the green shaded area. In order to make scores consistent, the scoring preprocessing (grey shaded box) is fixed and independent of the preprocessing used to construct the match.

4 Results

Several different variable matching preprocessing strategies were compared using the scoring method outlined in Section 3.3. For each strategy, DeepMatching homographies were computed for 500 image pairs. The different preprocessing strategies considered were: no preprocessing (unprocessed), Gaussian subsampling only (downsampled), and three levels of sequential preprocessing (light, heavy, and superheavy). The light sequential strategy applied the least contrast correct and denoising, while the superheavy strategy applied the most. As discussed in Section 3.3, before scoring, images are preprocessed using a fixed scoring preprocessing that is independent of the variable matching preprocessing used to derive the alignment. It was found that the accuracy of the scoring algorithm was increased by strong contrast correction and denoising of the input images. As such, the sequential superheavy preprocessing strategy was chosen as the fixed scoring preprocessing and so was used to process all images prior to scoring. Henceforth, ‘preprocessing’ refers specifically to variable matching preprocessing.

Given a score S returned from the DDSCORE as described in Section 3.3, an image alignment is defined as successful if $S > 0.5$, such that over 50% of changes in the images are improvements. Alignments with $S < 0.5$ are said to have failed. Finally, alignments with $S = 0.5$ are, by construction of the scoring algorithm, those where an insignificant number of pixel locations have changed either for better or for worse. The unscored alignments are made up of two cases, which at present are indistinguishable to the scoring algorithm. Firstly, a large fraction of randomly generated matches will be largely aligned from the onset, and so the score of 0.5 reflects the fact that the transformed image is not significantly different from the original (as no transformation needed to take place). Secondly, the images may be initially unaligned, but the matching does not

significantly transform the first image in the pair, and as such there will be very few pixel locations that are detected as having changed.

	Successful		Failed		Unscored
	Count	Score	Count	Score	Count
Unpreprocessed	15	0.8 ± 0.18	37	0.18 ± 0.15	448
Downsampled	19	0.8 ± 0.15	17	0.18 ± 0.19	464
Clustered	95	0.88 ± 0.14	17	0.4 ± 0.09	388
Sequential Light	79	0.9 ± 0.13	22	0.29 ± 0.16	399
Sequential Heavy	105	0.87 ± 0.16	19	0.33 ± 0.14	376
Sequential Superheavy	142	0.85 ± 0.16	24	0.27 ± 0.16	334

Table 1: The performance of five different preprocessing strategies for 500 KL11 image pairs matched using DeepMatching. Scores for each category are calculated by averaging scores across images that are in that category. Images in the unscored category by construction receive a score of 0.5.

The results in Table 1 clearly demonstrate the importance of variable matching preprocessing for successful image alignment. The number of successfully matched images increases from 15 in the case of unpreprocessed images to 142 for superheavy preprocessing strategy. Moreover, the low success rate (ratio of successful to failed alignments) of the unpreprocessed and downsampled preprocessing strategies, as compared with the other strategies, suggests that DeepMatching is not consistently performing well. Clustering was shown to perform worse than the heavier sequential methods. This could be due to the coarse binning of the images into five clusters such that each image receives preprocessing that is not specifically tailored for it. Meanwhile, the superheavy strategy yields the highest number of matches, and also has the highest success rate.

These results therefore suggest that DeepMatching performs better with heavy contrast correction and denoising applied to images. In order to best make use of the dataset of KL11 images for physics purposes, it might be decided to maximise either the number of images that are successfully aligned, or the quality of individual alignments. Because all the mean scores of the successful matches are similar, while the number of successful matches increases significantly, it seems reasonable to pick the prepossessing strategy with the highest number of successful matches.

Three examples of image alignments are shown in Figure 7. The scoring is likely to underestimate the success of the DeepMatching algorithm due to images containing plasma or other edge cases in which the scoring is not accurate. Discussions of further work that could be done to refine the scoring algorithm to improve performance are found in Appendix A.4.

5 Conclusions

An overview of a framework to automatically align tokamak images has been presented. The framework relies on the DeepMatching algorithm discussed in Section 3.1.2. A key part of the framework is a scoring algorithm, introduced in Section 3.3, that can automatically determine the quality of an alignment. This score allows the optimisation of the alignment process, including the preprocessing stages, as discussed in Section 2. In Section 4, different preprocessing strategies were shown to have a strong impact on the performance of the DeepMatching. The results suggest that DeepMatching performs best with heavily denoised input images that have had strong local histogram equalisation applied. The sequential preprocessing method outperformed the clustering approach.

The project has demonstrated the feasibility of the alignment of noisy diagnostic image data from tokamaks. The next steps are to investigate whether image alignment can be improved by further optimisation of the preprocessing. A fully automated search through the space of preprocessing parameters could be conducted. Additional work (outlined in Appendix A.4) is needed to refine and further demonstrate the reliability and accuracy of DDScore. The framework should then be validated using simulated distortions of real images, or alternatively an entirely synthetic set

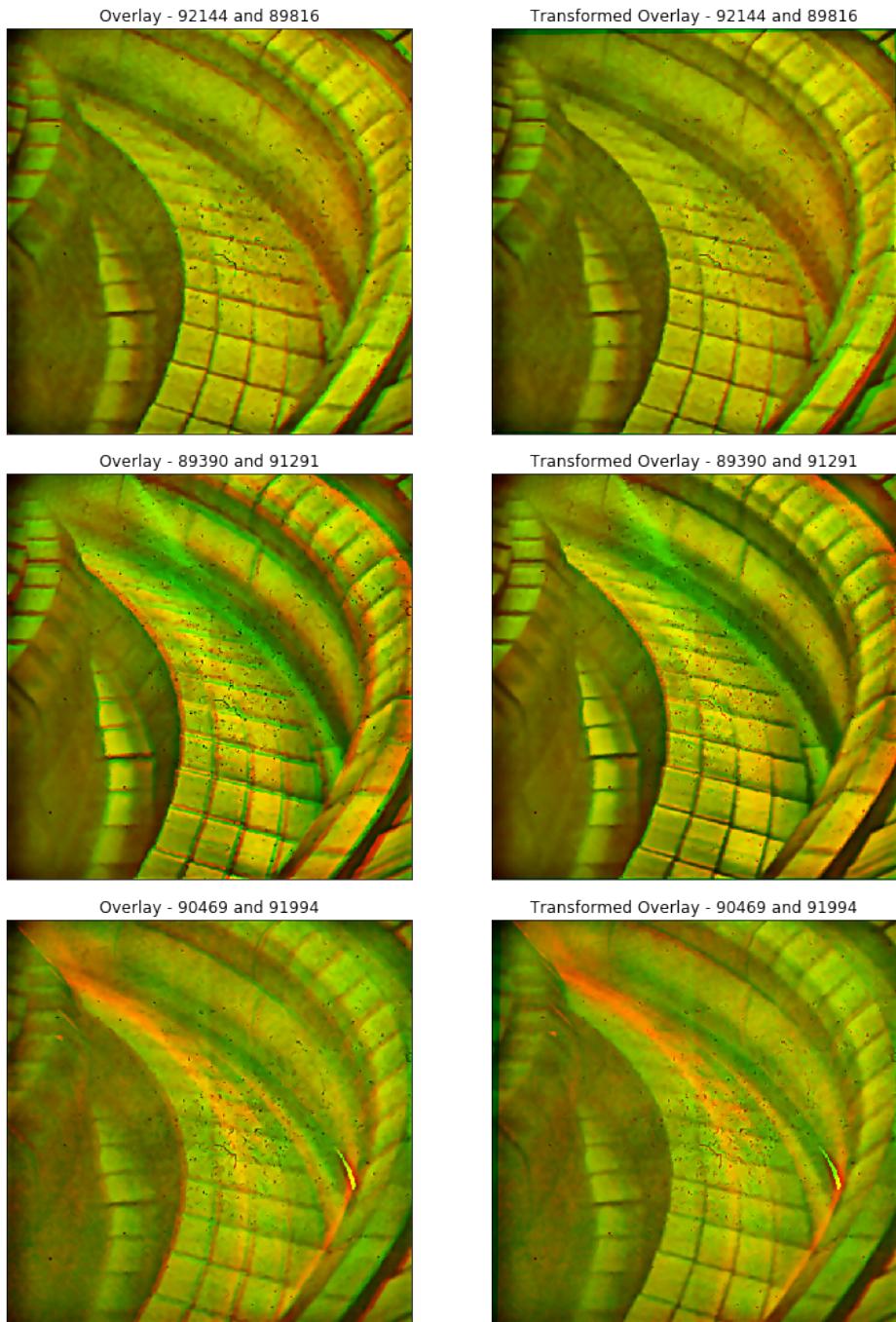


Figure 7: (top) An example of a poorly scoring match. Note the increasing width of the red and green fringes in the bottom right of the images going from left to right. This change outweighs modest improvements in alignment in other regions of the image. (middle) An example of a well scored match scoring 0.96. The images are successfully aligned. (bottom) A successful match that scores poorly at 0.49 due to the plasma in the image. Images are preprocessed before DeepMatching using the sequential superheavy strategy.

of images generated from the CAD models of the reactors. Video stabilisation is an area in which encouraging preliminary results were obtained, however further work is needed to fully demonstrate the methods suggested in Appendix B.1. Integrating an entire automated procedure into CALCAM, the software package currently used at CCFE for image analysis, would represent the final stage of the project. If fully realised, this would help to unlock valuable physics information stored in the data collected at MAST, JET, and future fusion reactors.

A Methods

A.1 Preprocessing Operations

The following sections contain detailed information about the various preprocessing operations used throughout the project.

A.1.1 Gaussian Subsampling

Gaussian subsampling is a parameter-free technique used to downsample images and thereby reduce the noise in the image. In performing Gaussian subsampling on an image, two operations are being carried out. The first is a smoothing step which takes a convolution of 4 pixels with a Gaussian kernel. The second is the actual subsampling of the image. Alternate rows and columns are discarded, such that the size of the image is reduced by a factor of two. This reduces the amount of smoothing present in the image at the expense of resolution. From Figure 8, we can see how this technique affects the raw images. The benefits from loss of noise vastly outweigh the downsides of the decreased resolution. Additional benefits are derived from the decreased computation time of further preprocessing operations, and of the subsequent matching process, due to the reduced size of the image.

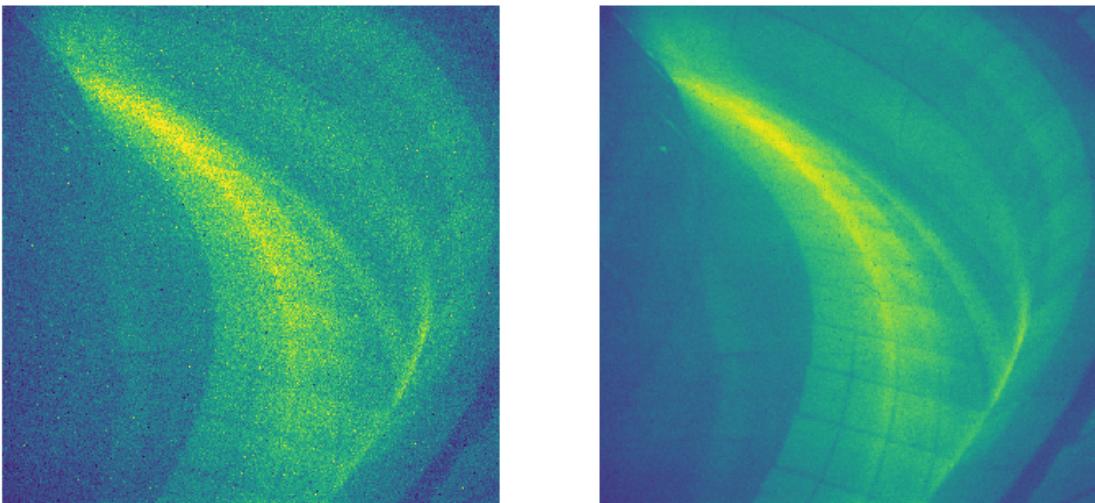


Figure 8: Before (left) and after (right) Gaussian subsampling of an image. Noise has been dramatically suppressed with minimal smoothing of the image. The subsampled image is half the resolution of the original image

A.1.2 Histogram Equalisation

An image can be converted into a histogram that shows us the distribution of pixel intensities in the image. Using the histogram, it is possible to measure the contrast and brightness of the image. For example, some images might have a very bright plasma present and this would appear as an increase in frequency at corresponding pixel intensity in the histogram, as compared to images with no plasma.

Of crucial importance are local histograms, which are histograms calculated for subregions of an image. Manipulation of local histograms in an image allows for contrast correction which is compatible with large natural variations in brightness and exposure across the image. A histogram equalisation tool was used called `cv2.createCLAHE` [2], which is a contrast limited adaptive histogram equalisation algorithm (CLAHE). The image is first split into tiles, and contrast limited histogram equalisation is applied to each tile. CLAHE depends on two parameters, the number of tiles the image is split into, and the clip limit for the histogram equalisation for each tile. Detailed

discussion of these parameters and algorithm for determining them automatically can be found in [7].

The `clip_lim` parameter controls how much equalisation is carried out. CLAHE is an important tool in normalising the contrast of images, so that they can be compared by a matching algorithm. While the tile size parameter was kept constant, the clip limit parameter was optimised in both the clustering and sequential preprocessing paradigms.

A.1.3 Denoising

Noise reduction was carried out using two filters, the bilateral filter and the non-local means filter. The bilateral filter takes an average of all pixels around the target pixel and filters out noise this way. A bilateral filter weights each pixel differently depending on edges or any depth detected in each image which helps smooth out unwanted noise but keep sharp edges. The parameters of the bilateral filter were kept fixed for different images.

The non-local means (NLM) filter takes an average of all the pixels in the image which is weighted to how similar each one is to the target pixel, this technique compared to a local filter results in less loss of information. The NLM filter depends on a strength parameter, which was optimised for, and a window size parameter, which was not. In general, a larger window size improves results of denoising at the cost of an increase in computation time.

A.2 Effect on Image Metrics

Figure 9 shows how the distributions of the metrics change before and after preprocessing. After preprocessing, noise is reduced, local contrast is increased, and images are more densely clustered. This demonstrates the success of the preprocessing step: the visual quality of images is improved, and the images are made to look more similar to each other. The noise parameter is calculated by subtracting a median smoothed copy of an image from the original and aggregating the differences between the original and resulting pixel intensities. The other metrics are measures of global and local contrast in the image.

A.3 Image Transformations

- *Fundamental Matrix.* The fundamental matrix transformation model generalises the homography matrix and allows relations of images with depth. Fundamental matrices are used in stereographic imaging.
- *Optical Flow.* Optical flow assigns velocities to objects in two different images based on the differences in detected locations of the object. It allows for fully non rigid transformations, such that objects in the image may move relative to each other. The authors of the DeepMatching package also have created an algorithm that uses the DeepMatching results to calculate an Optical Flow for an image pair. Though this technique was not tested, details can be found in [5].

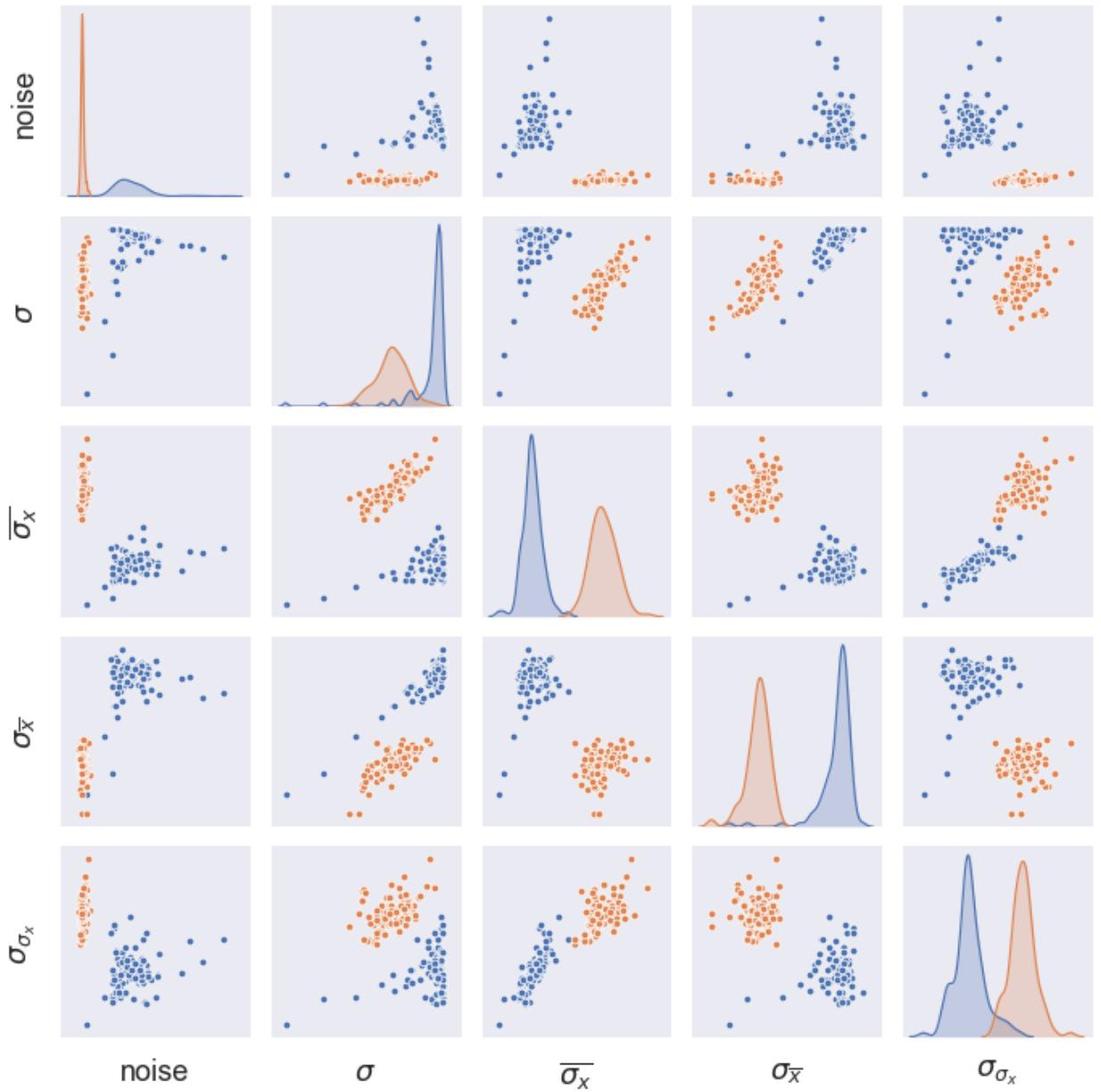


Figure 9: Downsampled version of the original images are in blue, while superheavy sequential preprocessed images are in orange. σ is the global contrast of the image. $\overline{\sigma}_x$ is the mean of the standard deviations of pixel intensities for each subblock in the image. $\sigma_{\overline{x}}$ is the standard deviation of the means of each subblock. σ_{σ_x} is the standard deviation of the standard deviations of each subblock.

A.4 Scoring

The next sections present ideas for the improvement and refinement of DDScore.

A.4.1 True Alignment Detection

There is currently ambiguity in the cases where DDScore detects no significant changes in an alignment (i.e. the algorithm returns 0.5). This is due to the fact that only changes in the alignment of the images are scored. DDScore cannot distinguish between an image pair that starts off perfectly aligned, and stays perfectly aligned, and an image pair that starts off completely unaligned, and stays completely unaligned. In order to resolve the difference between these two cases, it would be possible to perturb the homography matrix slightly and recompute the alignment scores for images transformed using the perturbed homography. If the initial homography was found to be at a maximum of the score, it could be inferred that the image is truly aligned.

This idea could be taken further still. If an improved version of DDScore can be shown to be sufficiently robust (perhaps using simulated data), it could be used to refine homographies from DeepMatching, or even generate homographies independent of DeepMatching. Preliminary success has been achieved by maximising DDScore over a small set of possible homographies in order to generate an alignment, however further work needs to be done to assess the flexibility of this technique.

A.4.2 Reducing the Impact of Noise

Due to dirt on the optics systems and other sources of noise, DDScore will often incorrectly track some pixels. Depending on the amount of variation between the images in the alignment, this noise could significantly affect the final score of the alignment. As shown in Figure 10, it is possible to significantly reject this noise by applying a median filter to the pixel locations detected as having changed. This should be done independently for the improved and deteriorated pixels. For a small radius parameter ($r = 3$), there is not a significant loss of signal, and noise is effectively suppressed. Further suppression of noise, at the expense of the loss of some signal, can be obtained by increasing the radius parameter further.

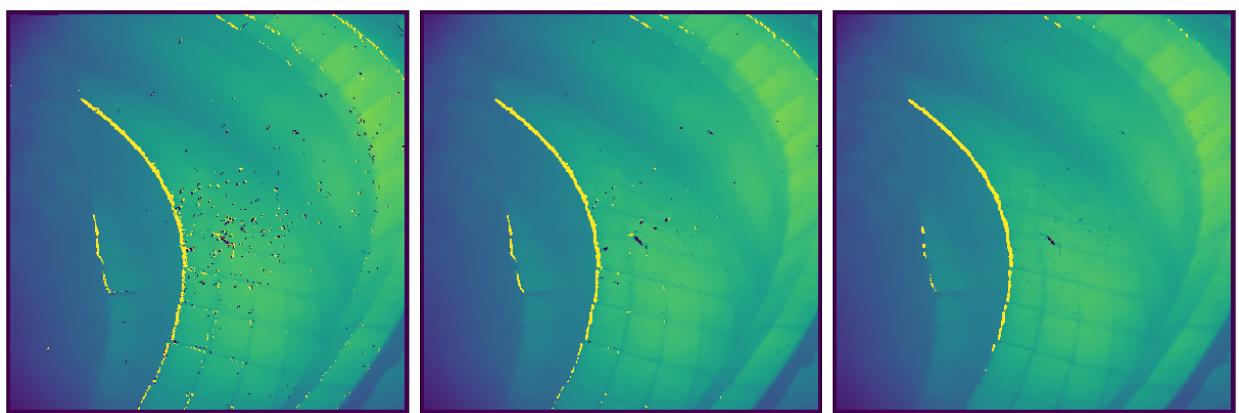


Figure 10: (left) the standard result of applying DDScore, showing significant salt and pepper noise in the central region of the image. (middle) The results of applying a median blur with a radius parameter of three to the pixels in the left hand image. (right) A median blur of the left hand image with a radius parameter of five. Pixels detected as having improved are coloured in yellow and pixels that have gotten worse are black.

A.4.3 Subblock Scoring

An alternative method of rejecting noise is to score the image in subblocks, and ignore subblocks in which only a small number of pixel locations are detected as having changed. Subblock scoring also allows for the correction of a large difference in the mean pixel intensity between $l_1^{(i)}$ and l_2 . Such large differences, in which a region of $l_1^{(i)}$ is much brighter or darker than the corresponding region in l_2 , can break the scoring algorithm, causing it to misclassify detected points. By making the mean pixel intensity within a subblock equal, these issues are mitigated.

A.4.4 Automatic Tolerance Detection

The DDScore depends on the absolute values of the pixel intensities in the input images through the tolerance parameter t , introduced in Section 3.3. For low contrast images, d^{ij} tends to be small, and as such a small value of $t \sim 10$ is appropriate. For heavily contrast corrected images, an appropriate value is $t \sim 50$. In order to make scoring consistent across images, even when comparing image pairs that have been matched using different preprocessing strategies, raw images are fed to the scoring algorithm after passing some fixed preprocessing step that is independent of the preprocessing used to generate the match (see Figure 6). This fixed preprocessing stage reduces the variability of contrast in the images, but, as shown in Figure 9, some variability still remains.

In order to ensure that the tolerance is set correctly for every image pair, the contrast metrics of the image could be used to set the tolerance. This could be achieved by taking a single image pair, and applying varying level of contrast correction to the images. Each resulting version of the pair could be scored, and then tolerances could be optimised based on the contrast metrics so that the scores are all consistent. Repetition of this process for several image pairs would build up a mapping from input image contrast to DDScore tolerance. Additionally, as images within a pair may not have the same contrast, the tolerance parameter need not be symmetric as it is in eq. 2. In this case, a tolerance parameter t_1 could be set based on the contrast of l_1 , while an additional parameter t_2 would be set from l_2 . A final refinement would be to set the tolerance on the subblock level, which could further improve the scoring accuracy.

A.4.5 Automatic Image Cropping

After applying a homography to l_1 , the resulting image l'_1 maybe warped in such a way that parts of it “fit inside” l_2 . For example, if the homography results in a translation of five pixels in the negative y direction of l_1 , then the top five pixels of the overlay of l'_1 and l_2 will contain only pixel intensities from l_2 (i.e. in this region $l'^{ij}_1 = 0$). If this region is presented to DDScore, it is likely to be scored negatively, as, if $l'^{ij}_1 = 0$, eq. 1 implies that d^{ij} will be negative. This artefact can quickly come to dominate over other legitimate sources of change in the images. To ignore border effects like these, images are cropped before they are scored by DDScore. Currently, images are cropped by 20 pixels on each side. However, for alignments which do not exhibit these features, cropping can remove signal from the alignment that would otherwise be accurately scored. It would be possible to improve the situation by masking the output d^{ij} using the regions where $l'^{ij}_1 \neq 0$ but $l'^{ij}_1 = 0$ (i.e. those regions l'_1 has been shifted out of by the homography. This technique, or similar, would replace the blunt cropping of images that is currently employed.

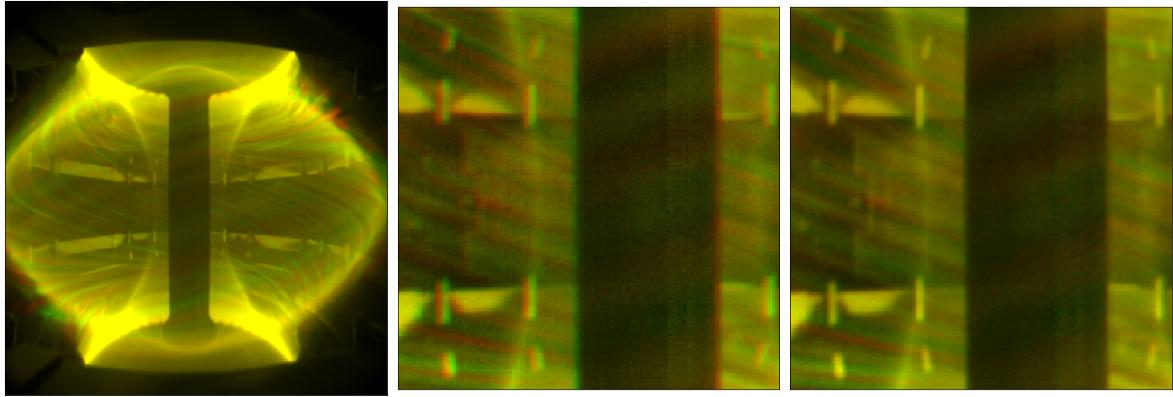


Figure 11: (left) Overlain frames of a pulse video in MAST. One frame is shown in the green channel, while the other is shown in the red. The bright plasma overlaps in both frames giving the image a yellow hue. (middle) A zoomed in section near the image centre. Red and green fringes are visible on the centre column and components in the background of the image. (right) overlain images after alignment using DeepMatching. The red and green fringes are gone, showing successful image alignment.

B Results

B.1 MAST Video Stabilisation

Additional data is in the form of videos of plasma pulses in the tokamaks. These videos last a few tens of seconds, during which camera motion is clear. The motion can be fast random shaking or slow drift. Figure 11 (left) shows a red-green overlay of two frames of a pulse video in MAST. These two frames are separated by 10 frames, which at a frame rate of 750 Hz equates to a difference of 13.3 ms. The red and green fringes around, for example, the centre column show how the camera has drifted. Distinct features such as this column are extremely useful to track when calibrating frame by frame.

Figure 11 (middle, right) shows an example of DeepMatching using a different camera view. The successful alignment shows that DeepMatching, along with homography generation through techniques such as RANSAC [6], is capable of rejecting the noise from the motion of the plasma in the images and tracking the static features. However, aligning an entire video using DeepMatching resulted in slow drift of the frame. To combat this, the centre column was used as a fixed tracking point that could anchor each frame to a known position. In order to track the centre column, the LSD feature extraction algorithm (Section 3.1.1) and parallel preprocessing techniques were used. After generating several copies of each frame by preprocessing an original frame in several different ways, line segment detection was run on each preprocessed frame. The lines from all the frames are then merged using a Hough transform [8]. Next, lines were filtered by gradients, keeping only vertical lines. The whole centre column could be tracked using two vertical lines. drawback of tracking only the centre column is that only motion in the horizontal direction is resolved. The preliminary results of this technique were positive, though additional work needs to be completed on a final implementation of the methods discussed here.

REFERENCES

- [1] *EUROfusion*. <https://www.euro-fusion.org/>. Accessed: 2019-04-02.
- [2] G. Bradski. “The OpenCV Library”. In: *Dr. Dobb’s Journal of Software Tools* (2000).
- [3] Anish Mittal, Anush Krishna Moorthy and Alan Conrad Bovik. “No-reference image quality assessment in the spatial domain”. In: *IEEE Transactions on Image Processing* 21.12 (2012), pp. 4695–4708.
- [4] R. Grompone von Gioi et al. “LSD: A Fast Line Segment Detector with a False Detection Control”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.4 (Apr. 2010), pp. 722–732. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2008.300.
- [5] Jerome Revaud et al. “DeepMatching: Hierarchical Deformable Dense Matching”. In: *arXiv e-prints* (June 2015). arXiv: 1506.07656.
- [6] Ondřej Chum, Jiří Matas and Josef Kittler. “Locally optimized RANSAC”. In: *Joint Pattern Recognition Symposium*. Springer. 2003, pp. 236–243.
- [7] Justin Joseph et al. “An objective method to identify optimum clip-limit and histogram specification of contrast limited adaptive histogram equalization for MR images”. In: *Biocybernetics and Biomedical Engineering* 37 (Jan. 2017). DOI: 10.1016/j.bbe.2016.11.006.
- [8] Nahum Kiryati, Yuval Eldar and Alfred M Bruckstein. “A probabilistic Hough transform”. In: *Pattern recognition* 24.4 (1991), pp. 303–316.