Key points detection algorithm for noised data

Yann Donon

This dissertation is submitted for the degree of Doctor of Philosophy in Data Science to
Samara National Research University
Faculty of informatics
Department of technical cybernetics

SAMARA UNIVERSITY

2020

Acknowledgments

This thesis was made possible through the encouragements and trust given to me by my family and friends, in Switzerland and Russia. I thank them for their presence from the bottom of my heart and dedicate them this work.

Special thanks go to the institutions that supported me through my work, in particular my supervisor and mentor at Samara National Research University, Dr. Sci. Alexander Kupriyanov and the head of the technical cybernetics department Academician RAS Victor Soyfer. Not to forget my research partner Dr. Rustam Paringer.

To the European Organization for Nuclear research (CERN), in particular the head of CERN openlab, Dr. Alberto Di Meglio, for giving me the opportunity to realize my thesis in their environment and enrolling me in the organization on that purpose.

To the organizations that brought me knowledge and inspirations during this work, the Image Processing Systems Institute of the Russian Academy of Sciences (IPSI RAS), that also accepted me among their ranks and its director Dr. Sci. Nikolay Kazansky. As well than the Progress Rocket Space Centre for their interest into the works presented in this thesis.

Last thanks go to an author who inspired me since my young age towards knowledge, both for humans and machine, and towards the stars.

"Violence is the last refuge of the incompetent."

- Isaac Asimov, Foundation

Abstract

This works introduces a new algorithm for feature detection in noised data, independently from the dimension of the given data. The algorithm is based on the detection and isolation of large features and its operability is demonstrated in this thesis through the development of two techniques, based on it. The first uses the algorithm for features detection on images, using image stitching as metrics for comparison with existing techniques. It demonstrates excellent performances on tests datasets registering a success rate almost three times higher than existing techniques while being fast and presenting a unique characteristic in the amount of points it detects for homography, largely inferior in number but superior in quality when compared to other techniques. The second technique demonstrate the performances achievable by the algorithm for feature detection on time series, it was developed in the framework of the SmartLINAC project at CERN. The technique showed excellent performance, detecting consistently all areas of anomalies, and labelling them correctly, where existing techniques showed large amount of false positive and false negative labelling entries due to the noise present in the data.

The algorithm's core concept is to ignore ambient noise in the data by a series of pre-processing techniques involving normalization, smoothing and thresholding, using noise's statistical distribution's attribute. Large areas are then isolated by blocks which's characteristics can be used for comparison.

The two techniques showed excellent performance in their range of application, proving the algorithm proposed in the thesis relevant and performant in its domain of application.

Contents

Introduction

Noised data represents a constant challenge for scientists, even when using modern technologies (Yanfang Li, 2008) however; such data exists all around us: whether on images as in drones and satellite captures, or images taken with damaged or partially obstructed objectives, meteorological conditions. It also exists in all sort of captors and time series, whether it depends on source volatility, environmental influence or humans' interactions. It can be stated that where there is data, there is potential noise. In small quantities, noise have little impact on existing solutions for features selection however, a growing noise results in a fast diminishing capability of those solutions. This represents a relevant problem as it leaves important quantities of data without solutions for features selection.

The need to solve the problem of features detection on noised data occurred to us twice in the recent years. One concerned image taken from satellites and drones, as because of the noise intensity present on the images, no known technique could detect the necessary features in order to stitch them, making them essentially useless. The second time, it appeared from a collaboration with the European Organization for Nuclear Research (CERN). The institution, together with the International Cancer Corp and the Science and Technology Facilities Council highly emphasized the need for simpler-to-maintain-and-operate medical linear accelerators (LINACs). But data received from the captors presented a noise intensity such that none of the tested techniques succeeded in detecting consistently and with an accuracy judged enough according to the situation the features necessary to analyse the signal.

As such, the scientific challenge presented is not only relevant as a novelty but in necessary to solve an engineering challenge. Tackling this challenge is meaningful for actors such at the aerospace industry but can also save lives when applied to areas such as medical LINACs. Key point detection is central to comparison between different data, whether it is to stitch images, compare sequences or changes, navigation , robotic mapping. This subject is therefore essential to advances in the domains mentioned above but not only, underlining the necessity to keep developing existing and new technique to maintain a state-of-the-art advance in this domain.

Existing techniques tends to focus on precise sequences in data series showing significant features, borders, angles or changes are used in most common techniques, however experiments presented below shows that those features are easily impacted the sources of noise introduced in this chapter, resulting in features selection failure.

This context shows the necessity to develop an algorithm, allowing to process and detect features on noised data from different sources.

The work presented in this thesis, is proven to be applicable on images and times series, while other applications are not demonstrated as they are out of scope, the research conducted, could be, be extended data as spectrometer imaging.

The novelty presented in this thesis is an algorithm which, when implemented, allows key point detection on noised data, it is demonstrated through this thesis that the algorithm proposed offers performances often higher than existing solutions, while preserving a unique functioning approach. Although the different elements presented in the algorithm are based on existing techniques, their assembly and order of operation was never used before and represents a major advance. This innovation proves itself relevant through the results obtained and the implementation of the techniques in the industry, where already existing techniques failed to perform in both cases, for aerial capture stitching in Russian aerospace and for anomaly detection at CERN, Switzerland, the algorithm proposed and its implementation were successfully implemented.

The core idea of the algorithm proposed in this thesis, is that noise is considered to be an integrant part of the data, that is should therefore not be isolated, treated, bypassed but just be taken as it is and spread according to a statistical approach, thus making it less relevant through its dissemination. Moreover, several key points, contained in a set with comparable noise statistical repartition, would see a similar repartition of the noise after filtering. The feature selection technique was from there adapted from this concept. Taking profit of this dissemination to isolate large, continuous key points. The combination of those idea in an algorithm and their application are the novelty the thesis is centred around.

1 Aim

The aim of the work presented in this thesis is to propose an algorithm capable of detecting key points in noised data from different sources. The algorithm should be implemented in the form of features techniques for the different data sources. The techniques should demonstrate and delimitate the capacities of the algorithm proposed and should bring a solution to the issues presented in the Introduction. The elements of the problems are the following:

Noise was shown multiple times as a challenge to key point comparison on images (Sarabjeet Kaur, 2017) (Zaragoza J, 2014) (Lu Y, 2018) (Nan Li, 2018), as shown in the mentioned articles, the issue has been approached several times, but no universal and highly performant solution stands out. In this context, experiences were performed in order to compare existing techniques when confronted to noise * (Y Donon et al, 2019). The techniques tested

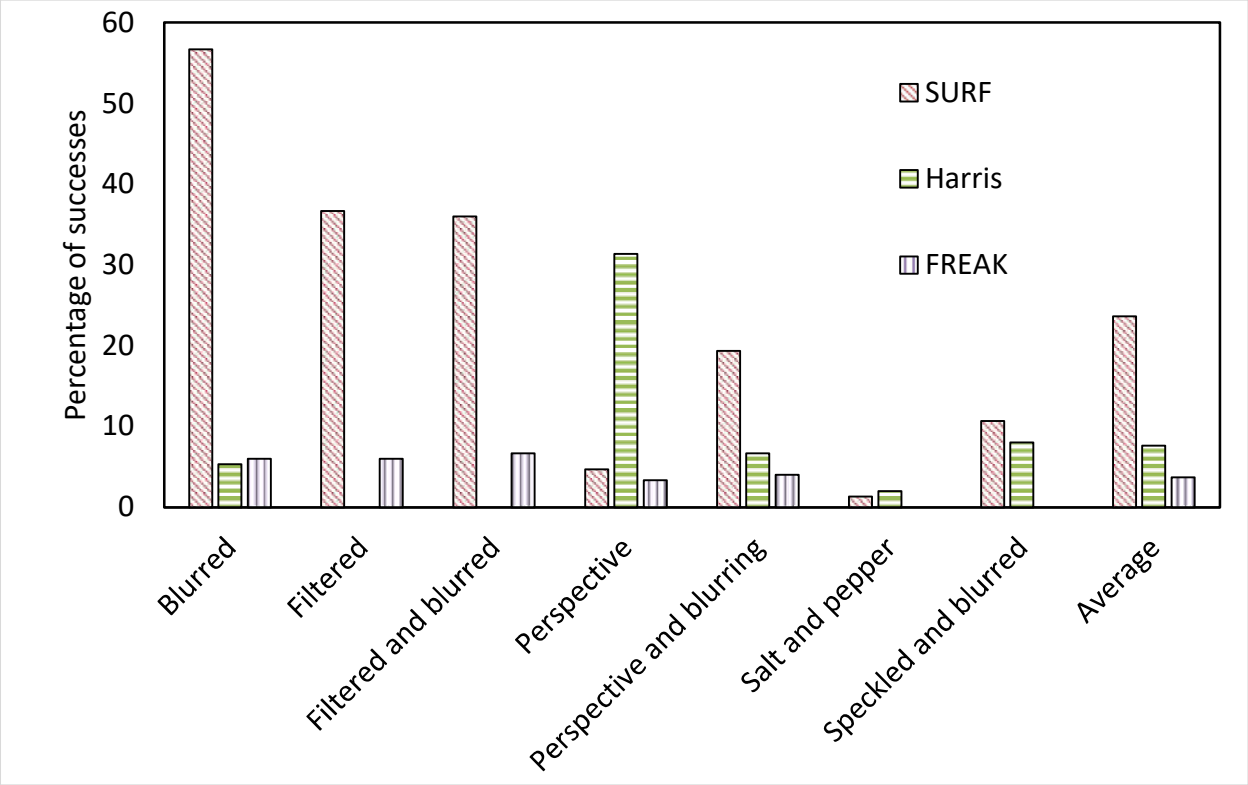


Figure 1 Success rate of Surf, Harris and Freak in stitching together images issued from a noised dataset described later ***Error! Reference source not found. Error! Reference source not found.****. *(Y Donon et al, 2019)*

were Surf, Harris and Freak, three reputable implementations, using different approaches for features detection. The techniques where confronted to different noise with increasing intensity, as a result of these experiment, the most successful tested technique (Surf) showed an average success rate of about 31%, which was at the time insufficient to solve the problem

of aerial picture stitching the research originated from. Figure 1 Shows the three-technique mentioned above and compare their success rate in stitching together images issued from a noised dataset as described later in 3.7 Results. Most noised image can be stitched with a success rate of 30% of higher, but no technique seems able to tackle all noise sources presented above.

Medical LINACs are complex platform used for cancer treatment (proton therapy, radiotherapy) and their use, especially in developing countries is highly impacted by frequent breakdowns and subsequent maintenance costs (David Pistenmaa, 2017). As such, the need for a breakdown prediction platform for linear accelerators was formulated into a joint project between CERN openlab and Samara University (Samara National Reasearch University, 2018). The experiment started using CERN's LINAC 4 RF power source data, but first analysis showed their nature, noised to the extreme was making analysis using existing technique challenging and results lacking of precision * (Yann Donon A., 2019). Figure 2 highlight imprecisions of existing techniques detailed in * (Yann Donon A., 2019) when compared to manual data labelling.

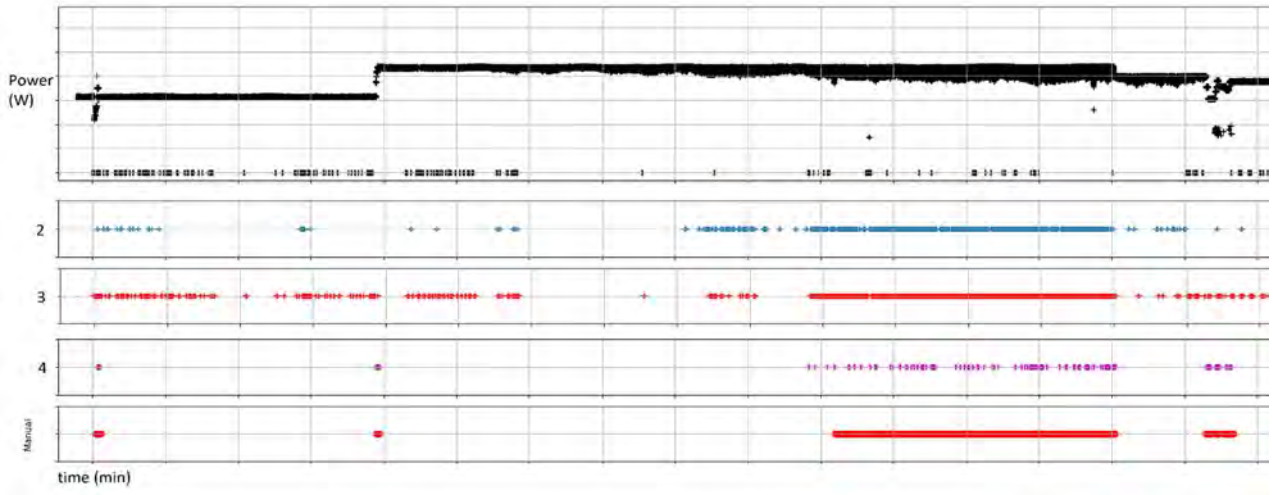


Figure 2 LINAC 4's data anomaly detection using existing techniques (Labelled 2, 3, 4), compared to the original labelling. All techniques shows significant imprecisions when compared to the original labelling.

The same algorithm should solve both problems presented in this chapter, through different implementations relative to the data source presented.

2 Tasks

As they are described above, it is understandable that although both paradigms are significantly different (images and time series), the problems encountered are in themselves very similar. As both problems appeared at the same period, they got often compared and preliminary work to solve both has been done in parallel. During this phase of pre-study, it was observed that once represented graphically, the approach of humans taking part to the project to understand both data sources was similar: looking at the data from a certain distance trying to guess general shapes that would differ from their environment. It is indeed a very human approach to be able to understand general patterns, or features, by taking a global picture, occulting details. This approach doesn't only apply to vision, finding a way even in therapeutic approaches in "evenly-suspended attention" technique (Freud, 1912). As such and as existing techniques hadn't met problem solving expectations it was decided to implement a technique to interpret image using the same concept creating the basis for the first technique developed in this thesis. Presenting excellent results at first, the technique used was broken down and adapted re-used for preliminary studies on the time series presented second in this thesis, again registering unique performances. From this point, the technique was refined and became a research axis creating the background for this thesis. As such, the following thesis develops the successful deployment of two techniques developed on the algorithm introduced on Figure 3.

The approach chosen to solve both problems is based on a single algorithm involving the same set of operations adapted to different data paradigms, one being images and the second time series.

The algorithm presented hereafter, which is the core of this thesis and the subsequently developed were constantly kept as simple as possible both in terms of workflow and in terms of engineering, while keeping them close to the original idea of mimicking human's behaviour when confronted to the original datasets.

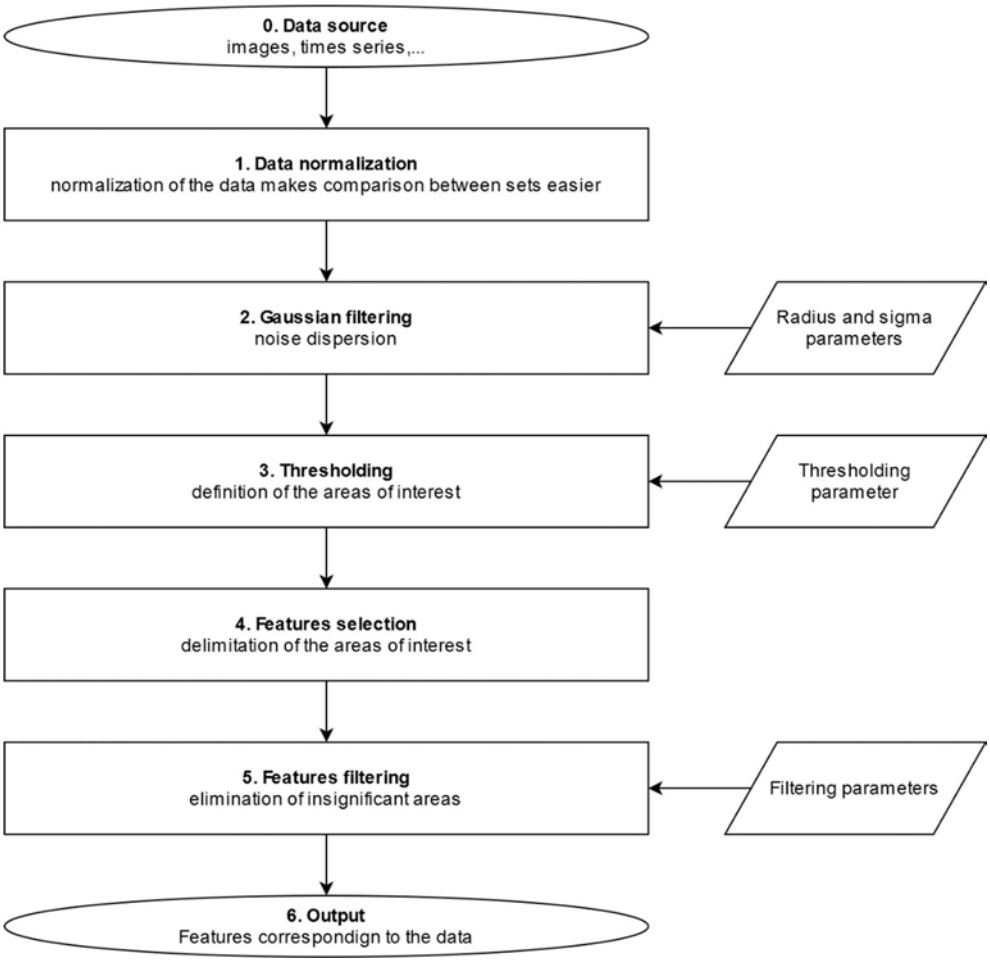


Figure 3 Representation of the algorithm presented in this thesis independently from the type of data that is to be treated.

As defined above, two engineering issues revolves around the algorithm presented. No tested existing technique managed to solve the mentioned problems in a way that could be considered successful (anomalies identified with enough confidence; noised images consistently stitched). The problems having to be addressed are:

- Aerial picture stitching (noised images).
- Anomaly detection in LINAC 4's plasma source (time series)

Both problems can be solved using the identification of key features in the data; the idea behind this thesis is to find an algorithm allowing the detection of key features in the different datasets and to two techniques allowing features comparison and treatment. As such the aims is to develop two technique allowing key points identification and comparison on noised data. The techniques should be similar based on the algorithm presented above, thus demonstrating its polyvalence. The resulting techniques should demonstrate significantly higher performances than other existing techniques when confronted to noised data, while maintaining a comparable resources consumption.

Tested on a noised images dataset, existing techniques succeeded at best in 23% of tested images * (Yann Donon A., 2019), the technique implemented in the framework on the thesis should reach a success rate of twice higher or more on the same dataset, or twice more results than obtained so far.

In the same way, exiting techniques manage to identify jitter on CERN's LINAC 4 data with 79% of confidence. The technique developed in our experiment should allow a confident superior to 95%.

- Developing a procedure allowing key points identification with high precision on noised images.
    - Compare the efficiency of this technique on different datasets. The objective is to obtain more than 46% success rate in image stitching on the noised dataset used to compare existing techniques. The reference number was selected as it represents a success rate twice higher the currently best performing technique.
    - Compare the technique's metrics (accuracy, processing time, success rate) when solving a stitching task (used as a reference for key points comparison).
    - Measure the technique efficiency different noise conditions (different noises types and intensities).
- Developing a procedure allowing key points identification with high precision on noised time series.
    - Compare the efficiency of this technique on CERN's LINAC datasets (more than 95% success rate in noise identification).
    - Compare the efficiency of this technique's metrics (accuracy, success rate) to identify anomalies on the given data (used as a reference for key points comparison).

2.1 Contributions

The contributions offered in this thesis are centred around the algorithm presented in Figure 3, the algorithm was developped in the framework of this thesis, and the objective is to prove its relevance. Past the tasks definition, the thesis shows the functioning and performances of the algorithm proposed when applied to images and time series. The contributions are structured as following:

- Description of the problem of key point detection on images
    - Description of existing techniques

    - Description of the process developped on the base of the algorithm proposed for key point detection on images (Blurred Image Matching, BIM)
    - Comparison between BIM and existing techniques
        - Performances
        - Specific features
    - Conclusion over the problem of key point detection on images
- Description of the problem of key point detection on time series
    - Description of existing techniques
    - Description of the process developped on the base of the algorithm proposed for key point detection on images (Series with Noise Featuring, SNiF)
    - Comparison between SNiF and existing techniques
        - Performances
        - Specific features
    - Conclusion over the problem of key point detection on time series
- Thesis conclusion taking in account the two processes, around the algorithm proposed.

In short, the innovation proposed in this thesis is a new algorithm for key point detection on noised data, supported by two implementations of this algorithm for key point detection on noised images and noised time series.

3 Key point detection on noised images

As described earlier, the aim of this thesis is key point detection on noised data, with an emphasis on two different sources, one being noised images. This chapter introduces the problem, starting by describing existing techniques, followed by the solution developped in the framework of this thesis and a description of the results obtained.

The objective is to find on two images $I(x, y)$ and $I'(x', y')$, sets of points (x, y) on I and (x', y') on I' corresponding to the same real-world object. With a minimal dependence to the prior presence of noise on the image. As such the mapping $(x, y) \rightarrow (x', y')$ researched should satisfy consistency constraints and minimize the energy cost represented $E(I, x, y, I', x', y')$ as in its simplest form:

Equation 1 image point matching minimized energy cost

$$E = \| I(x, y) - I'(x', y') \|$$

3.1 Existing techniques

The initial need, causing the development of a key point detection technique for noised images followed the presentation of a dataset of aerial captures that were to be stitched. Those captures presented noise to a degree making existing technique to consistently fail performing this given task. The first test dataset used was from drone captures in mid altitude and presented

Figure 4 Example of aerial view image including meteorological conditions induced noise, blur and partial lenses obstructions.

mainly three categories of noises : meteorological conditions induces noise, blur and partial obstruction on the lenses as represented in Figure 4.

Nowadays, most commonly used and implemented feature detection algorithm includes corner and edge description (Harris, FAST) or feature description (SIFT, SURF, FREAK) (Pooja Ghosh, 2015) (Gary Bradski, 2008). Other algorithms exists and are commonly used as BRIEF, or ORB, but they are alternatives to already mentioned techniques, BRIEF is an alternative for SWIFT taking less memory, ORB stands for “Oriented FAST and Rotated BRIEF”, which speaks for itself (OpenCV, 2019).

In the framework of this research, Harris (Accord.net, 2019), SURF (Accord.net, 2019) and FREAK (Accord.net, 2019) (Alexandre Alahi, 2012), were selected for comparison based on their reputable implantation in the Accord.net framework (Souza, 2014). Consistency of some key results have also been controlled using a similar implantation using the OpenCV framework (Rustam Paringer, 2020).

All those techniques succeeds to identify at least a pair of matching points however, the amount of irrelevant points is too important for the comparison algorithm described in 3.7 Results to perform a correct filtering. Which leads directly to the feature comparison to fail.

The three following sub-chapters describes the existing techniques tested in the framework of this thesis.

3.2 Harris corners detector

Chris Harris and Mike Stephens developed an early corner detection attempt. The technique is based on the research of displacement intensity for displacement. It focuses on image regions containing texture or isolated features, combining corner and edge detection. The technique was first developed for natural imagery in 1988 and is still commonly used. (OpenCV, 2019) (Chris Harris, 1988)

Figure 5 Shows feature detection using the Harris operator on the image “Lena”. It is noticeable that as the technique description indicates, points are mainly found in corner or edges (intersections), such as illustrated in the focus.

Figure 5 Shows an image on which Harris feature detection has been applied. White dots represents features selected by Harris as significant for a comparison. Focus on the images shows three points locations characteristic of the technique.

Figure 6 shows an application of Harris corner detection on the image introduced in Figure 4. The image illustrates the limits of the techniques, on the enlargements showed the left of Figure 6 three points are situated around a building, two of them are valid, showing that the technique is not completely disabled. However, one of them is calqued on a lens's partial obstruction point. In this image, most features found are located on such points, as illustrated in the enlargement on the right. This results in the technique being unable to perform to identify features with enough precision of an algorithm such as RANSAC.

Figure 6 Shows a Harris implementation for features detection, using the Accord.net framework. The image shows that the techniques holds on camera lenses' partial obstruction, and detects no points in the clouded area.

3.3 SURF

SURF is essentially a speeded-up version of the SIFT algorithm (Lowe, 2004). It was developed in 2006 and uses a Hessian matrix measurement detector and a distribution-based descriptor (Haar wavelet sums around points of interest (Jin-Sheng Guf, 1996)), after transforming the image using Laplacian of Gaussian approximation (Herbert Bay, 2006). However, SURF simplifies existing techniques and process. SURF was first described as outperforming existing techniques in terms of repeatability, distinctiveness and robustness. (OpenCV, 2019)

Figure 7 Shows feature detection using SURF on the same image than previously. Orientation on interest point (green lines in circles on Figure 7) are determined by the largest sum of Haar wavelet responses of size 4σ in a radius of 6σ of the detected point of interest. The circle's colour depends on the Laplacian sign, blue if negative, red if positive (Evans, 2009). Features are mostly found on areas where the image is gradually changing as illustrated Figure 7.

Figure 7 Shows an image on which SURF feature detection has been applied. Circles' center represents points selected by SURF as significant for a comparison. Focus on the images shows three points locations characteristic of the technique.

The Figure 8 shows an application of SURF corner detection on the image introduced in Figure 4. It is understandable from the image that the technique is heavily influenced by the noise present on the image. In particular in the sample selected on the left, where when compared to other samples, where a similar feature can be found, following the “cloud”. The sample enlarged on the right on the image shows on the other hand shows the how the point of interest orientation can be wronged by lenses obstructions. As previously, some of the features found are relevant and would allow operations based on their however tests showed they were insufficient in this context to perform a RANSAC operation with an acceptable success rate.

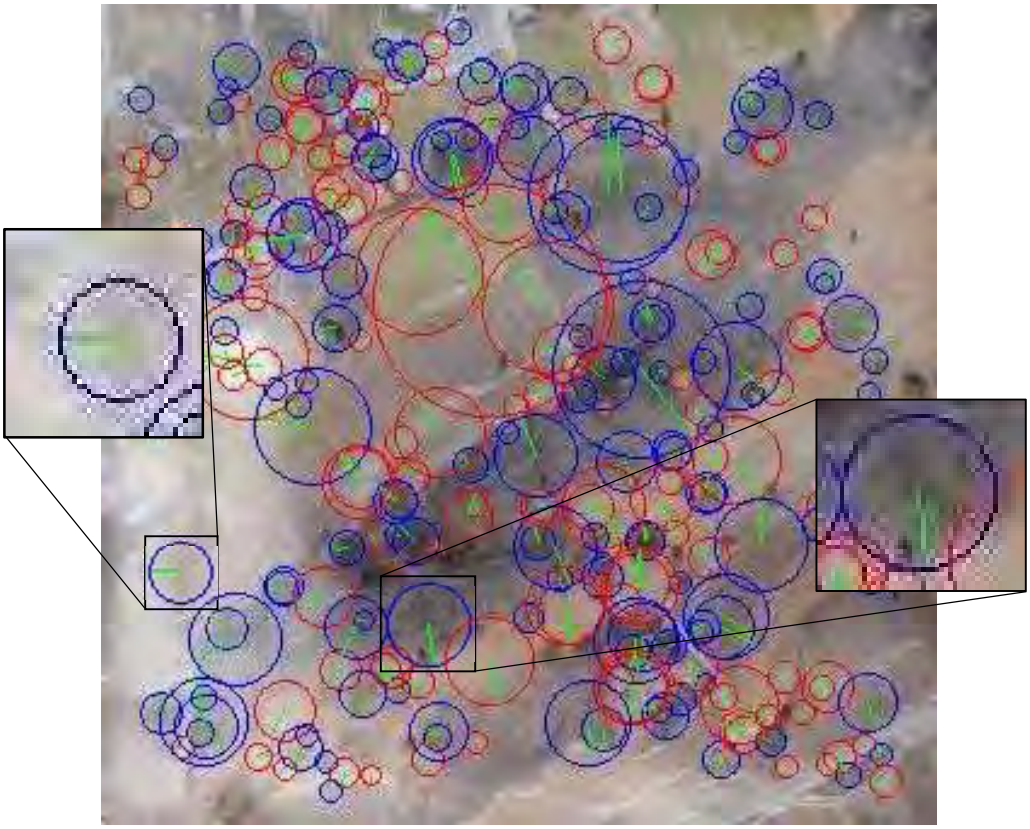


Figure 8 Shows a SURF implementation for features detection, using the Accord.net framework. The image shows how the technique is influenced by noise.

3.4 FREAK

Fast Retina Key-point (FREAK) is a descriptor inspired by human retina visual system. It computes a cascade of binary string using a retinal sampling pattern. The technique was developed in the “Ecole Polytechnique Fédérale de Lausanne (EPFL)”, in Switzerland in 2012. It was developed specifically for embedded applications and tested by its authors as in general more robust and faster to compute, with a lower memory consumption than SIFT, SURF and BRISK. (Alexandre Alahi, 2012)

Freak uses a coarse-to-fine descriptor (Hantao Yao, 2016) and successive cascade of comparison based on human retina sampling system (Alexandre Alahi, 2012). Figure 9 shows points distribution after an image analysis using FREAK, points are mostly located on extremities and borders.

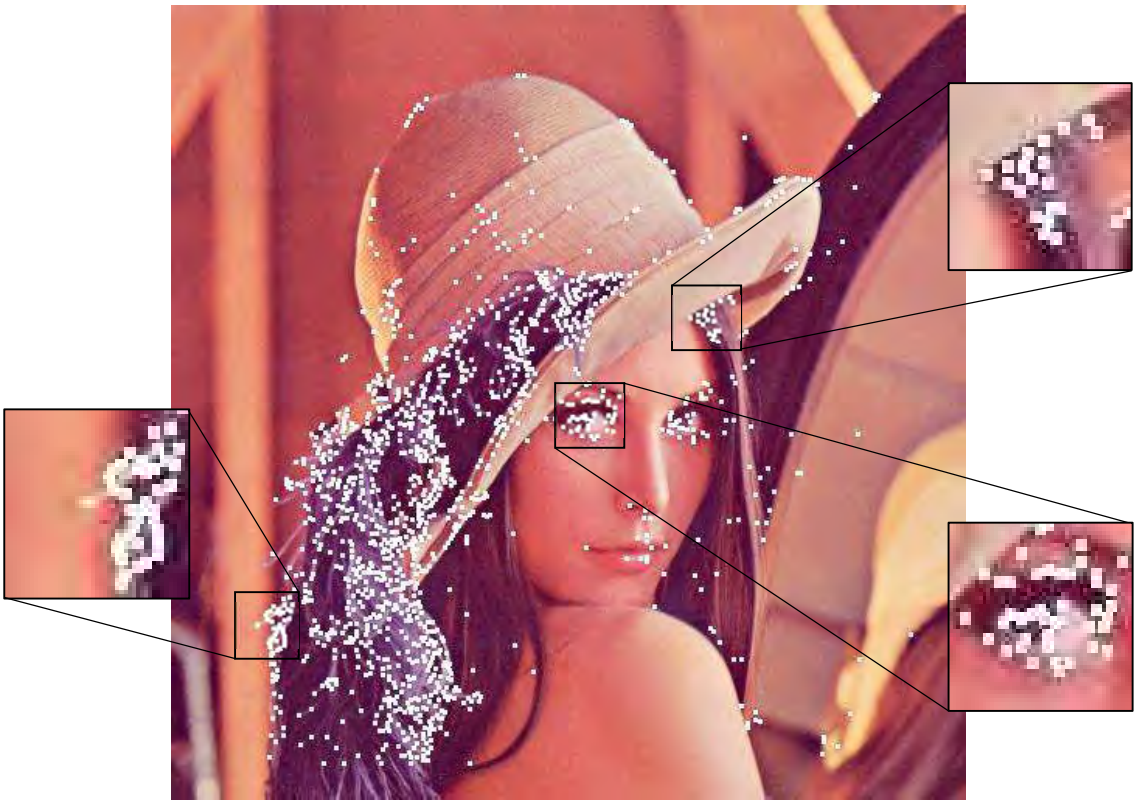


Figure 9 Shows an image on which FREAK feature detection has been applied. Circles’ center represents points selected by FREAK as significant for a comparison. Focus on the images shows three points locations characteristic of the technique.

Figure 10 shows an application of FREAK corner detection on the image introduced in Figure 4. The features points representation is very similar than the one presented above in 3.2 Harris, the most obvious difference to the human eyes is the amount of features found, significantly greater than from the technique mentioned above. The enlargement presented on the left of the image shows the same area than on Figure 6, presenting the same characteristics, even if more points are present in this case, they tends to be too influenced by noise present on the lenses. Comparative analysis made on the points found in the clouded corner are insignificant as they depend entirely on the noise in the area. The enlargement presented on the left show 3 points, one of them only being attached to a noise sample. Again however, the accurate features although existing, didn't allow the RANSAC algorithm to perform with enough confid

Figure 10 Shows a FREAK implementation for features detection, using the Accord.net framework. The image shows although it is less visible than in Figure 6 Shows a Harris implementation for features detection, using the Accord.net framework. The image shows that the techniques holds on camera lenses' partial obstruction, and detects no points in the clouded area that the techniques holds on camera lenses' partial obstruction it moreover still detects few points in the clouded area.

3.5 Blurred Images Matching

Blurred Image Matching (BIM) * (Y Donon et al, 2019) * (Yann Donon A. K., 2019) * (Yann Donon R. P., Brightness normalization for Blurred Image Matching, 2020) * (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020) * (Rustam Paringer, 2020), unlike techniques presented previously detects features (blobs) on the images, ignoring the noise through a serie of preprocessing described further in 3.6 BIM process. Resulting in performances competing with other techniques on non noised sets but clearly overtaking their performances when dealing with noised sets as presented further 3.7.3 aResults evaluation. Figure 12 shows blobs detexted on the same image presented earlier in this chapter. Unlike on the iamges presented in Figure 6, Figure 8 and Figure 10, noise in completely ignored in BIM results, in comparison with Harris, SURF and FREAK. Instead the technique locate a large area (blob) in th middle of the image.

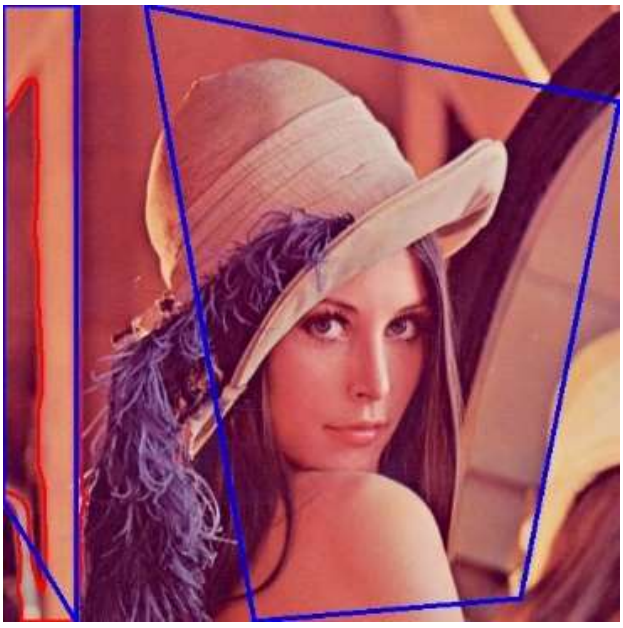


Figure 11 shows feature detection on Lena, as introduced in Figure 5, Figure 7 and Figure 9s. Three blobs are visible according to the processing steps described further in this chapter (one represented in red on the left and two in blue).

Figure 12 echoes with Figure 4, presenting BIM-s blob detection after the pre-processing steps presented in 3.6 BIM process. This specific feature detection detects 5 points, which is sufficient for a homography, as in the case of this image all the points are usable.

3.6 BIM process

Although already mentioned technique presents in most cases very good performances, BIM takes a different approach than its predecessors, the technique is based on image pre-processing and blobs detection. The reason behind this radical change in approaches is that BIM was primarily developed with for objective to detect features on noised images, therefore pre-processing steps are oriented towards noise compensation.

3.6.1 Process presentation

The following chapter presents an example based on two noised images to be matched together through their features. The matching evaluation is made using stitching in this research the two images do not present level of noise that wouldn't make it possible for other techniques to stitch them and are only used as a process explanation. Figure 13 shows the original images used to demonstrate the stitching process.

Figure 14 shows BIM's image stitching process, as detailed in chapter 3.6 BIM. With a numbering corresponding to the one introduced in the algorithm, presented in Figure 3.

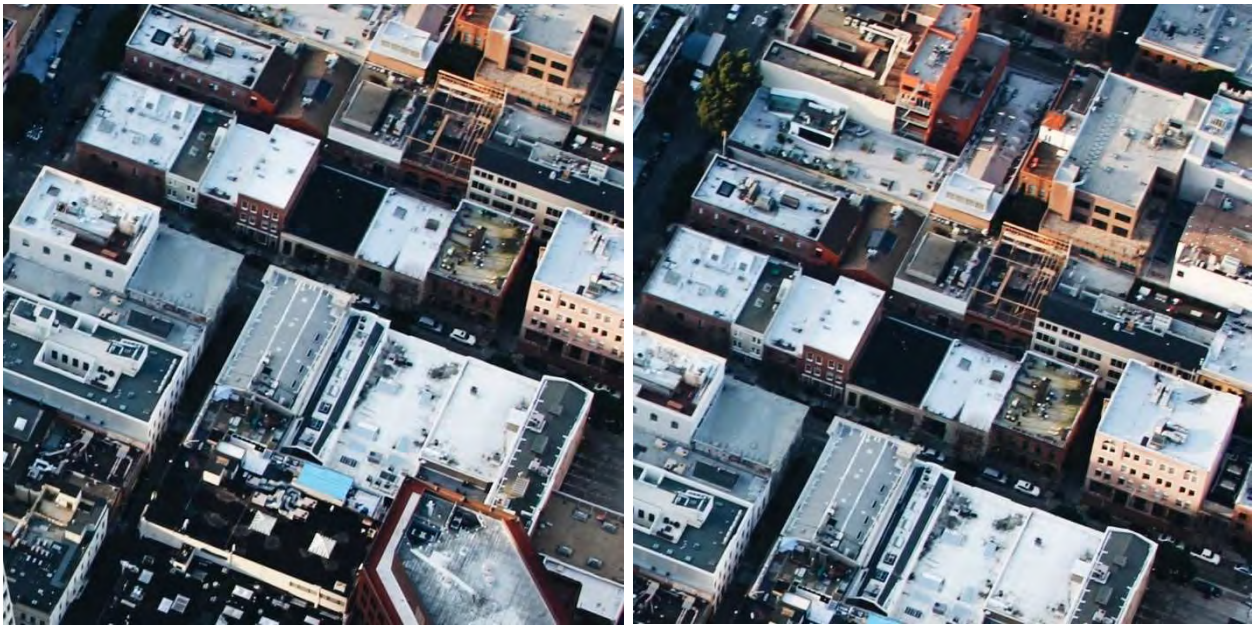


Figure 13 The two original images used to demonstrate BIM's stitching process.

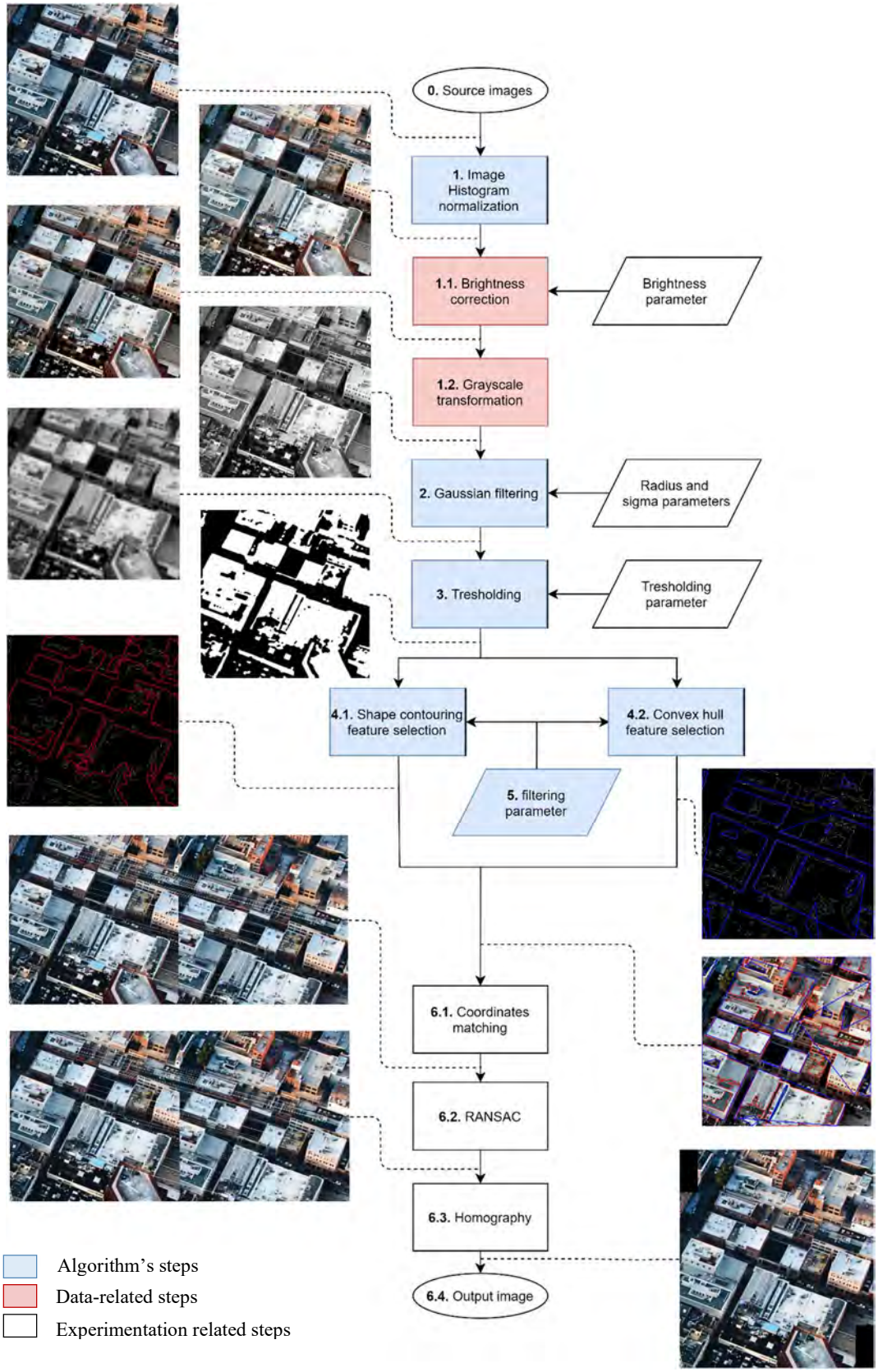


Figure 14 Illustrated BIM process used for image stitching.

3.6.2 Histogram normalization

Histogram normalization (algorithm's step 1 as on Figure 3 and BIM process' step 1 as on Figure 14) consists of changing pixel's intensity range on an image, resulting in generally augmenting image's contrast or to create consistency in datasets. This achieves two desirable effect for BIM, it first attenuates the effect of noise due to brightness differences between images, and second, it allows a more pronounced distinction of features, which is useful to distinguish blobs. Histogram normalization can be done on both grayscales and coloured images. In the second case, colour histograms (red, green, blue) are separated before the same steps are applied than for grayscale histograms. BIM uses histogram normalization on coloured images. The formula used, where the image I, the image minimum and maximum intensity, respectively Min and Max, the resulting image minimum and maximum intensity, respectively $newMin$ and $newMax$, corresponds to the following: (Rafael C. Gonzales, 2007)

Equation 2 Image histogram normalization

$$I_N = (I - Min)\frac{newMax - newMin}{Max - Min} + newMin.$$

BIM's first step consist of the image brightness comparison and correction. As showed in chapter 3.6.3 Brightness correction, the technique's performances are influenced by the image brightness. A brightness difference exceeding 3% leads according to experiments results in significantly lower quality results.

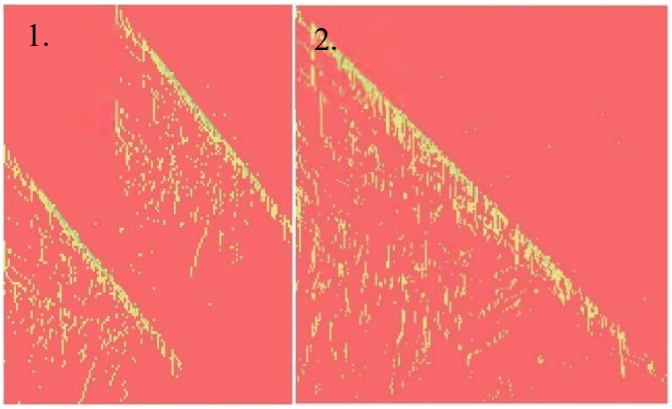


Figure 15 Results of brightness correction and matching between a pair of images. The closer the color is to green; the more points were found with a combination of brightness correction. 1) a pair of images before histogram normalization, 2) the same pair after histogram normalization.

Without histogram normalization, it is possible to match images presenting different level of brightness by finding which combination of corrections are the most efficient, however, as shown later in 3.6.3 Brightness correction, this technique places the images two different correction spectrum, depending on their brightness, as illustrated in Figure 16 * (Yann Donon

R., Brightness normalization for Blurred Image Matching, 2020) shows two images 1.a. and 1.b. going through the BIM stitching processing. Their original histogram distribution shape (RGB distribution) looks similar (2.a. and 2.b.), as both images are taken in the same terrain, but their brightness level is widely different, making the histogram being distributed on different sides of the RGB spectrum. In this use case, the normalization is applied on each colour plane separately. After normalization (3.a. and 3.b.), the images present the same average brightness (by design, 128 as histogram normalization uses mean brightness). The histogram has been flattened, resulting a contrast augmentation, which shows helpful as it

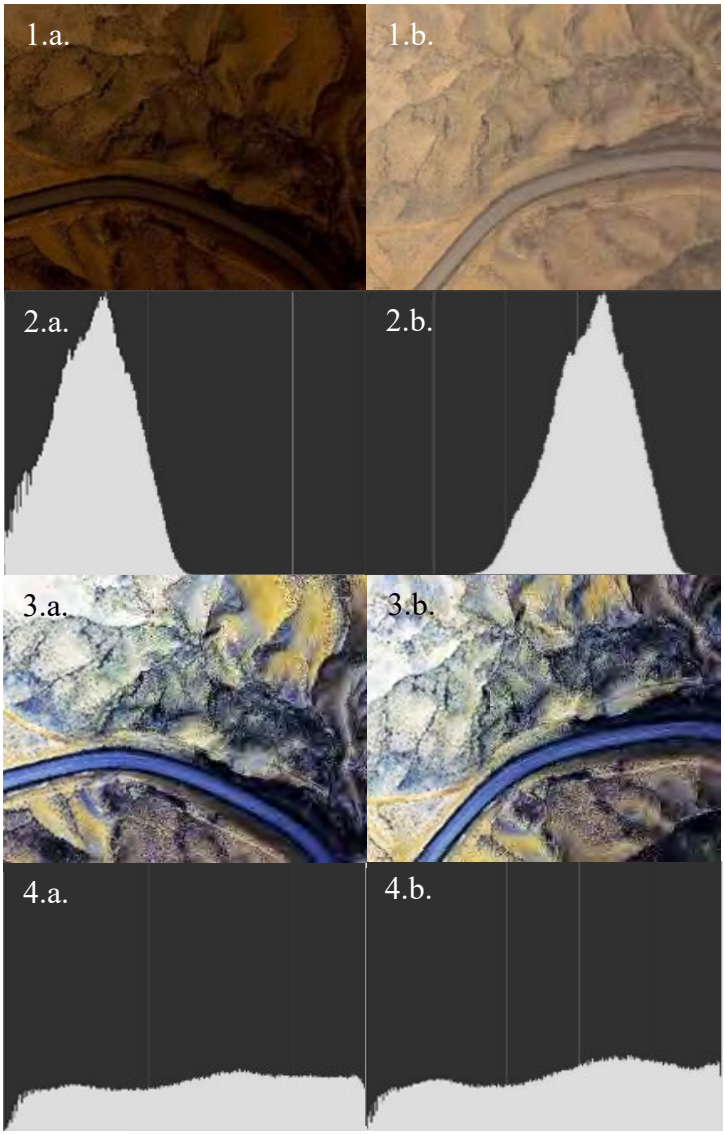


Figure 16 Process of image brightness normalization and the image's histogram before and after processing (Yann Donon R., Brightness normalization for Blurred Image Matching, 2020)

increases BIM's found amount of shapes by an average 87% when compared to the same image, with the same brightness, before normalization. * (Yann Donon R., Brightness normalization for Blurred Image Matching, 2020)

Figure 17 shows the images introduced in Figure 13, after histogram normalization processing. The result is less flagrant than in Figure 16 as the colour repartition in Figure 17 (wide predominance of red and green) is from the source wider than in Figure 13, resulting in fewer changes when flattening the histogram.

Figure 17 Images presented in Figure 13 The two original images used to demonstrate BIM's stitching process after Histogram normalization.

3.6.3 Brightness correction

Difference in brightness between images is a source of noise that must be taken in consideration. This step is not described in the algorithm (Figure 3) as it is relative to the data source and the feature selection process only, it corresponds to Figure 14's step 1.1. BIM's pre-processing steps, in particular the one described in 3.6.6 Thresholding is very sensitive to brightness difference, experiences showing optimal performances when the brightness difference remains under 5% * (Yann Donon R., Brightness normalization for Blurred Image Matching, 2020). Average brightness is arguably pixel's most significant characteristics, yet no standard formula exists for its measurement. In this thesis, colour vector length mean arithmetic model was used (Sergey Bezryadin, 2007). Where: Br, the average brightness, n, the number of pixels in the image and r, g, b the pixel value in the RGB spectrum:

Equation 3 Image average brightness

$$Br = \frac{1}{n} \cdot \sum_{i=0}^{n} \frac{(r_i + g_i + b_i)}{3}.$$

As such, for an image I, it is possible to determine if normalization is necessary if the following statement is false:

Equation 4 Normalization necessity calculation

$$\left|\frac{Br_{I1} - Br_{I2}}{Br_{I1} + Br_{I2}}\right| < 0.05.$$

Experiments were performed in * (Yann Donon R., Brightness normalization for Blurred Image Matching, 2020), a pair of images were selected and their brightness corrected in a range of ± 256. Registering for every combination the amount of points matched between the images, resulting on a graphic in two parallel diagonals, one for every combination of brightness correction and its opposite (for example for an image having a difference of 20, (-10:10) and (10:-10)) , with a distance relative to the image's brightness difference as showed on Figure 18.

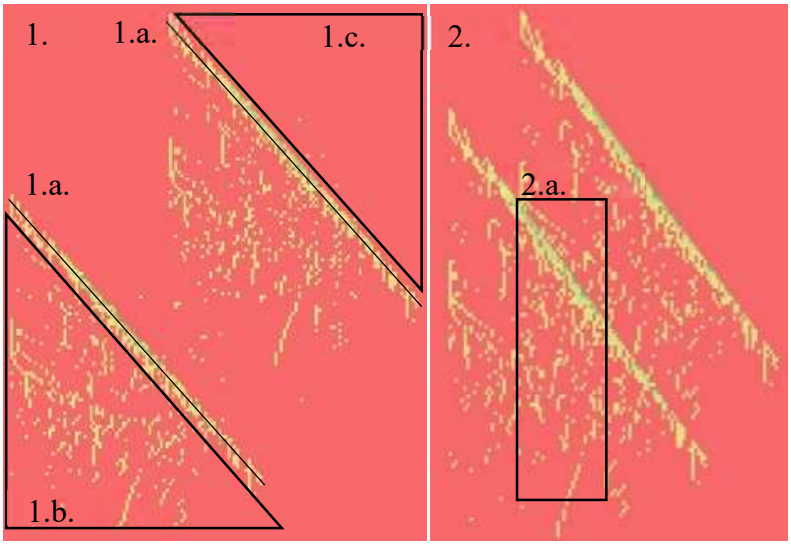


Figure 18 Difference matching results' histogram. On the left, the image's average brightness difference is 120% higher than on the right. The original image is the same in both 1. And 2.

As stated in * (Yann Donon R., Brightness normalization for Blurred Image Matching, 2020), on Figure 18, the area directly below the diagonal Figure 18.1.b) shows dispersed points; those are either noise or isolated points that are very distinctive shapes on an image. Brightness change has a lesser impact on such shapes; they are usually caused by a sudden change of colour in the landmark (such as a red roof in a green forest). The area directly above the diagonal (Figure 18a.1.c) is empty as it represents the part of the array where images are brightness correction of both images diverge in opposite directions; any point is this area is almost certainly noise.

Experiments represented in Figure 18 also showed that all points were found in a brightness level ranging from 30 to 202, with a steady peak in ± 0.25σ, or between a brightness of 89 and 113 * (Yann Donon R., Brightness normalization for Blurred Image Matching, 2020). Figure 19 shows the average of points found on a dataset depending on the images' average brightness. As stated above, histogram normalization increases the average amount of point found in an image, however it is important to notice that the amount of points found in the ± 0.25σ of the brightness distribution curb is the same.

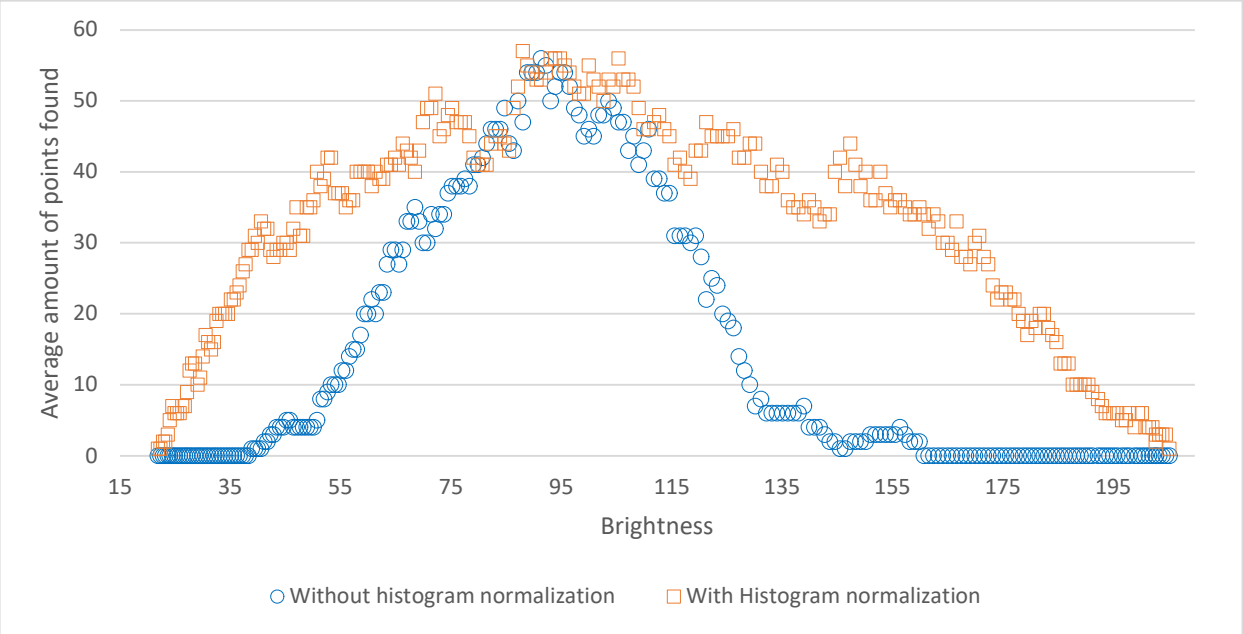


Figure 19 Represents the distribution of points found with and without brightness normalization. The curb using brightness normalization has a significantly higher variance. (Yann Donon R.).

BIM's default has accordingly been set to 113 as it is the closest value in the ± 0.25σ to the mean brightness applied by histogram normalization, 128, thus minimizing changes.

Figure 17 shows the images introduced in Figure 13, after histogram normalization and brightness correction processing. The average brightness difference between Figure 17 and Figure 20 is -15, bringing the image from 128 to 113. The difference can be considered little for the human eye but it above the 5% recommended for an optimal use of BIM.

Figure 20 Images presented in Figure 13 The two original images used to demonstrate BIM's stitching process and Figure Figure 17 Images presented in Figure 13 The two original images used to demonstrate BIM's stitching process after Histogram normalization. after brightness correction, here the images presents an average brightness of 113.

3.6.4 Grayscale transformation

Grayscale transformation was used in order to simplify and thus accelerate the blurring process and thresholding process. As such, this step is not described in the algorithm presented on Figure 3 but it corresponds to Figure 14's step 1.2. Accord.net framework was used to perform this action, using the BT.709 algorithm for transformation with the following coefficients (Accord.net, 2019):

- Red: 0.2125.
- Green: 0.7154.
- Blue: 0.0721.

ITU-R Recommendation BT.709 is a well-known algorithm for grayscale transformation standardized in 1990, it is commonly used for HDTV systems. (International Telecommunication Union, 2015)

Figure 21 shows the images introduced in Figure 13, after step 1 to 3 as represented on Figure 14.

Figure 21 represents the images introduced in Figure 13 The two original images used to demonstrate BIM's stitching process after Figure 14 Illustrated BIM process used for image stitching's steps 1, 2 and 3, Grayscale transformation using BT709 algorithm.

3.6.5 Gaussian blurring

The way BIM uses to treat sets of noised images is not to approach the noise specifically in any way, but it a way to "drawn it in more noise" * (Y Donon et al, 2019). Noise in negated, using Gaussian blurring, this step corresponds to the algorithm's (Figure 3) and BIM's process (Figure 14) step 2. Gaussian blurring is a filter based on the Gaussian function (Weisstein, 2019) and its application on an image pixel for transformation. The pixel's characteristics transformation for a two dimensional transformation is the following where x, y are pixels coordinates on an image and σ the Gaussian distribution's standard deviation (Shapiro Linda, 2001):

Equation 5 Gaussian blurring transformation

$$\mathrm{G}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

According to this formula a convolution matrix is applied to the image in which each pixel's value is recalculated according to a weighted average of the pixels neighbouring them in a radius depending on the σ value. The closer the pixels from the convoluted pixel, the higher its weight. In our formula, the σ value is calculated in function of BIM's optimal kernel size (ksize) according to the function * (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020):

Equation 6 Gaussian sigma calculation as used by OpenCV

$$\alpha = 0.3 * \big((ksize - 1) * 0.5 - 1\big) + 0.8.$$

Used as default parameters by OpenCV Gaussian blurring function (OpenCV, 3).

Figure 22 illustrate how details disappears in their surroundings, blending into larger blobs. It is those blobs that are used by BIM as features as described in 3.6.7 Shape contouring blobs comparison and 3.6.8 Convex hull blobs comparison

Noise, just like details of an image is blend into larger shapes. Smaller shapes as seen on the bottom left of Figure 22 represents noise for BIM's applications, on this image, shapes are too small and too precise (therefore sensitive to change) to be compared with precision. However, on the bottom right image, large and characteristic blobs can be observed, it is that kind of blobs that can be compared between each other.

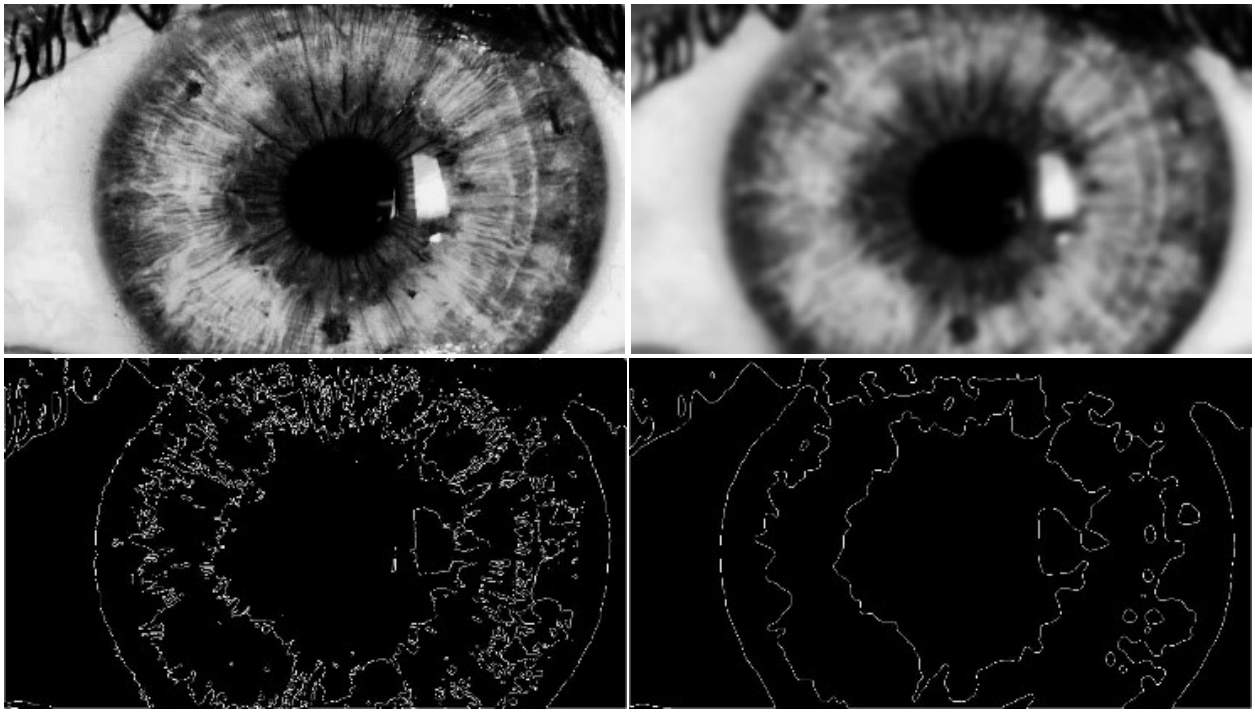


Figure 22 Examples of Gaussian Blurring on an eye image with for both image their counterpart after thresholding and edge detection.

The necessity for the technique to present larger shapes is confirmed by Figure 23, in this figure, the same pair of images are compared, with different threshold values (Y axis) and different blurring values, starting from 0 (X axis). Red indicates the absence of matches between images, yellow its presence and boxes going towards green indicates more positive matches. On the figure, images presenting no or little blur before comparison are represented on the right of the image, the further to the right, the more features matches between the two

images, thus highlighting the need, up to a certain degree, to apply blurring to the images in BIM's process.

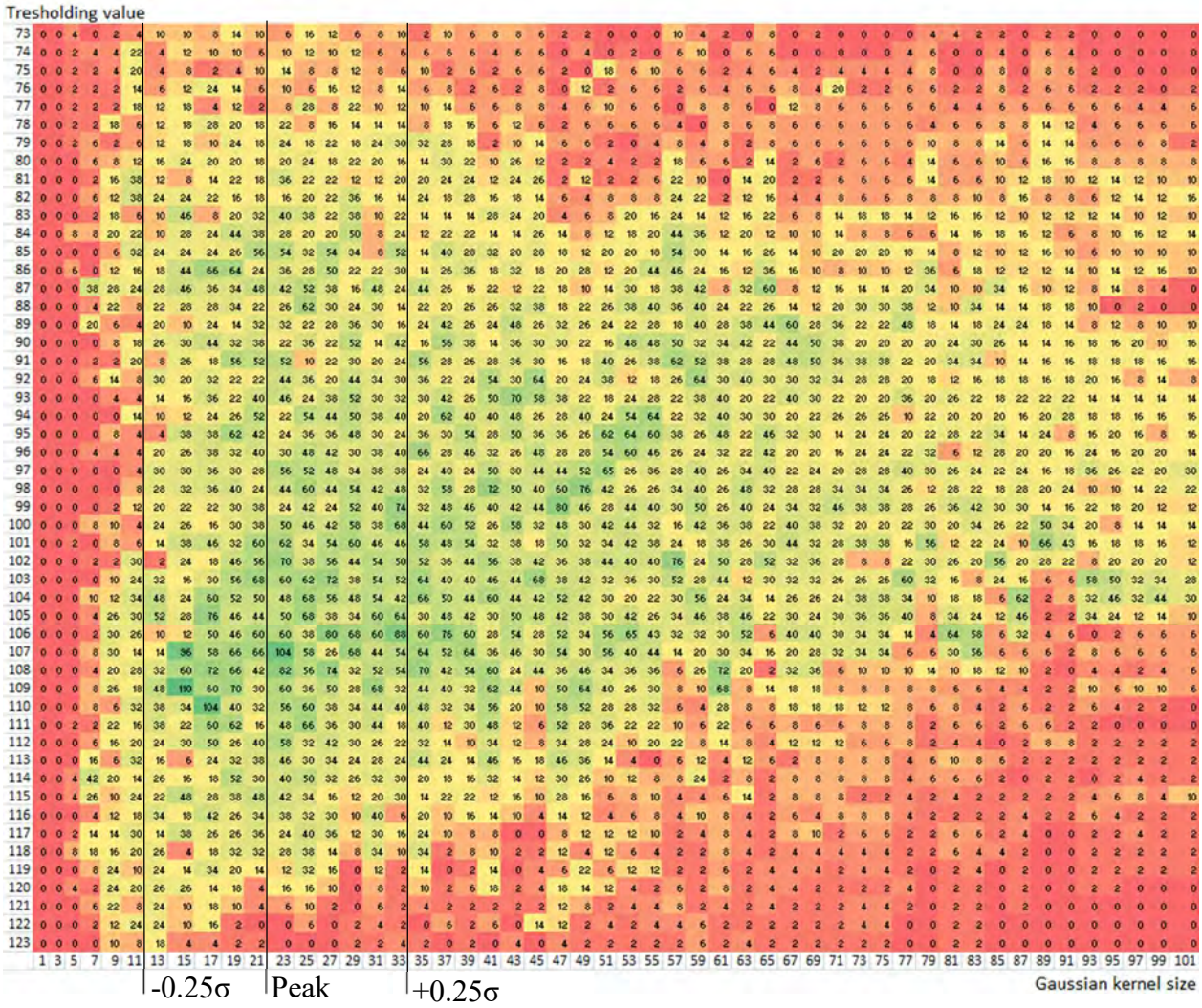


Figure 23 Partial blob match count between two pairs of images depending on Gaussian Kernel size (X) and Thresholding value (Y). Overlaid, the statistical peak of point found on a dataset, as described later in the document.

Experiments presented in a previous article shows that the ideal amount of point found is at its peak with a Gaussian kernel size of 21 and a ± 0.25σ range of statistical repartition between 12 and 30 * (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020). Figure 24 illustrates the amount of points found depending of the Gaussian kernel size. Consequently, in the absence of specific parameters, 21 is used as the default Gaussian kernel size.

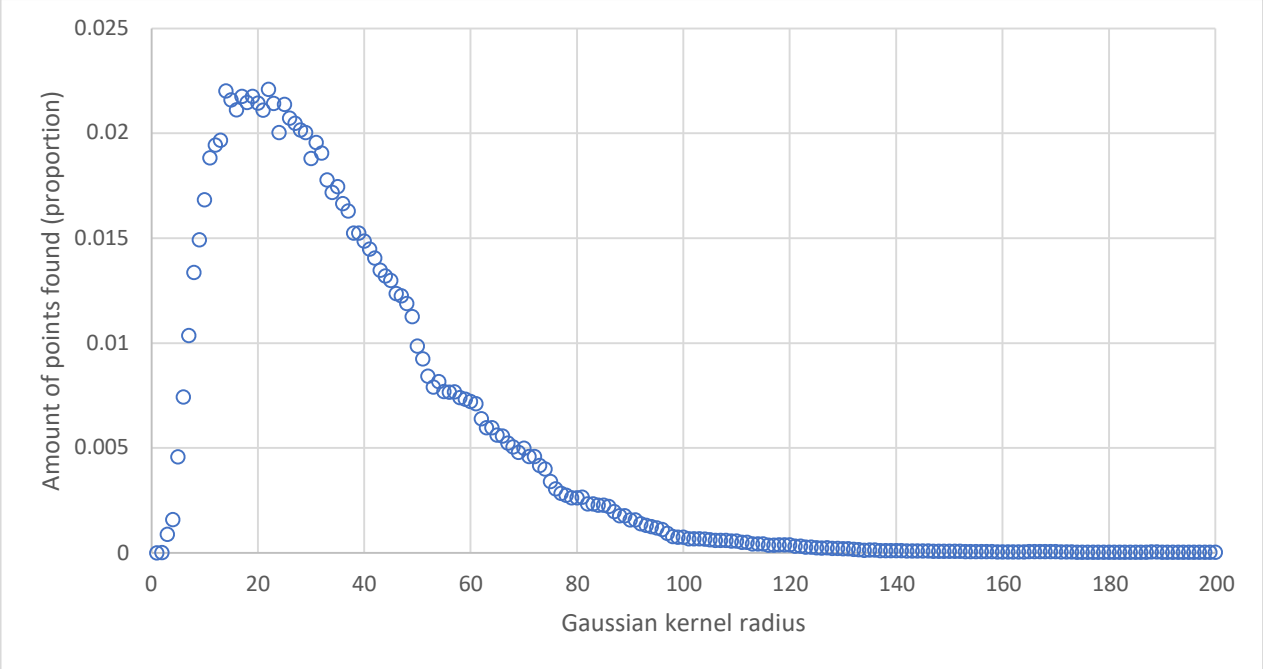


Figure 24 Statistical repetition of points found depending on Gaussian kernel size.

Figure 25 illustrates the pair of images illustrated earlier in 3.6.1 Process presentation after Gaussian blurring operation, with a kernel size of 21 and the sigma calculation presented above.

Figure 25 represents the images introduced in Figure 13 The two original images used to demonstrate BIM's stitching process after Figure 14 Illustrated BIM process used for image stitching's steps 1, 2, 3 and 4, Gaussian transformation with a kernel size of 21.

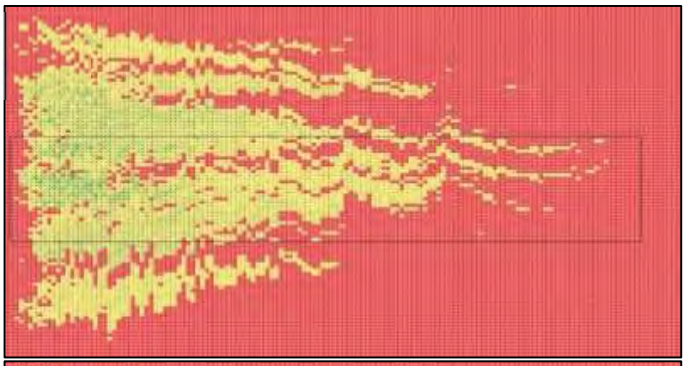
3.6.6 Thresholding

Thresholding is a way to create a binary image (black and white) through a process of image segmentation. It is used in BIM to reform blobs after the Gaussian blurring. It corresponds to the algorithm's (Figure 3) and BIM's process (Figure 14) step number 3.

Thresholding operates according to an image's pixel intensity and a constant T, the thresholding value. For every pixel $I_{i,j}$, for $I_{i,j} < T$ the pixel is changed to black and for $I_{i,j} > T$ to white. This operation is essentisal for BIM's functioning and yet a very sensible point as described in * (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020)

Thresholding depends is relative to individual colour channels brightness, Figure 26 shows the recapitulation of images being matched by blobs depending on different Gaussian kernel radius (axis X)and thresholding values (axis Y). Figure 26.a. shows the recapitulation when all the image's channels are compared (R,G,B) , Figure 26.b. shows a recapitulation from the same source images but when only one colour channel is selected a distinctive part of the Figure 26.a appears (highlighted in the image). As such, every recapitulation is separated in three sources, specifically visible in high threshold value ranges thus making difficult to select an ideal threshold value.

a.



b.

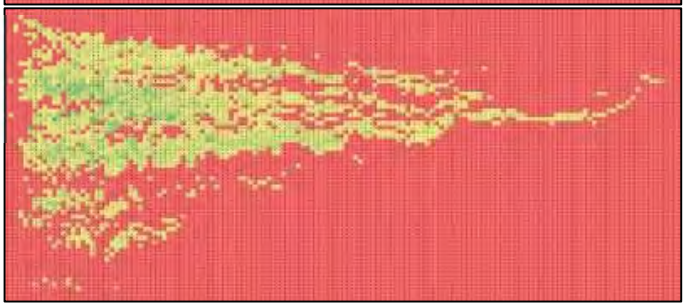


Figure 26 a. represents the matched points array of an image. B. Represents the same image matched points when the blue canal only is retained on the image. (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020).

These issues' impact is however greatly diminished by histogram normalization as described in chapter 3.6.2 which fixes a standard brightness for each pixel. Figure 27 highlights these three channels, in the pair of images matched, the blue channel is predominant, creating "trenches" as highlighted in the image.

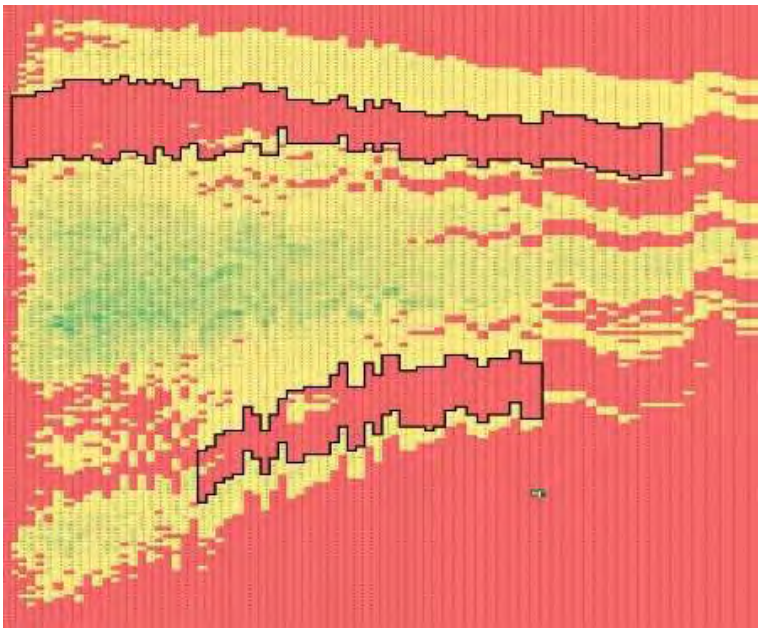


Figure 27 estimated representation of trenches between color channel pikes. It is noticeable that both sides of the trenches fits together closely. (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020).

This situation is explained as some blobs with uniform colour falls either above or under the threshold value. As the Gaussian blurring has for effect to make uniform some areas, they can fully disappear over a single threshold value difference as illustrated in Figure 28.

Figure 28 Same image with different threshold values of one on the right part of the figure. (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020).

It is possible however to define a preferred thresholding value, as for the brightness and blurring value, a range have been selected, corresponding to the peak's ± 0.25σ of blobs matches on an example dataset, or a threshold value between 101 and 115 with a mean as 109, which is selected as a default value for BIM.

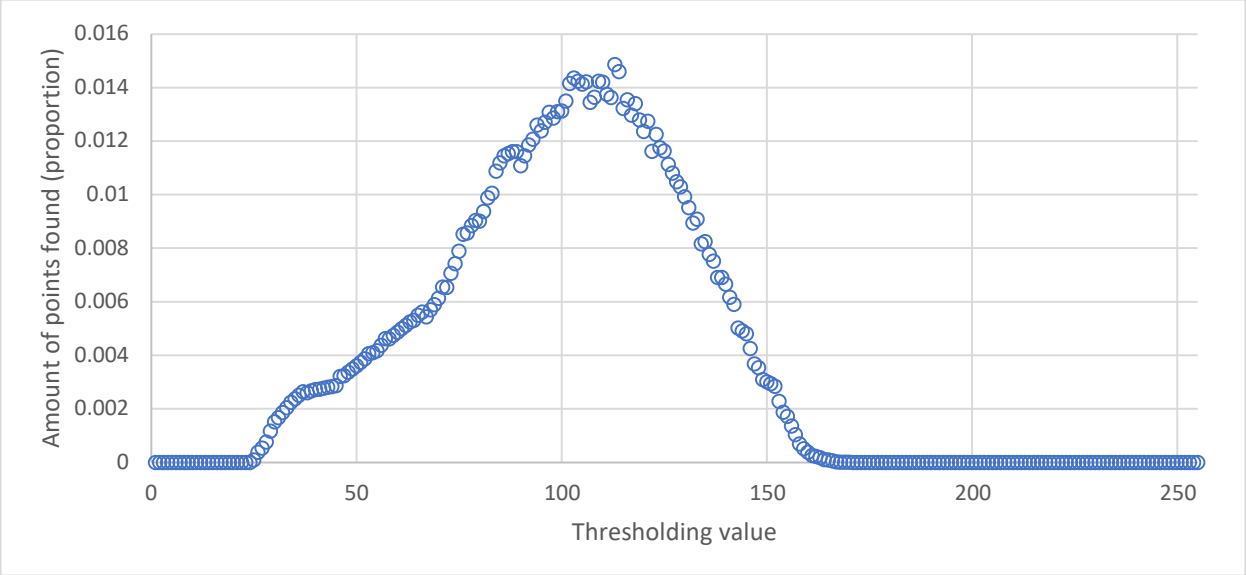


Figure 29 Statistical repetition of points found depending on Threshold value. (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020).

Figure 30 shows the reference image introduced in 3.6.1 after the thresholding process.

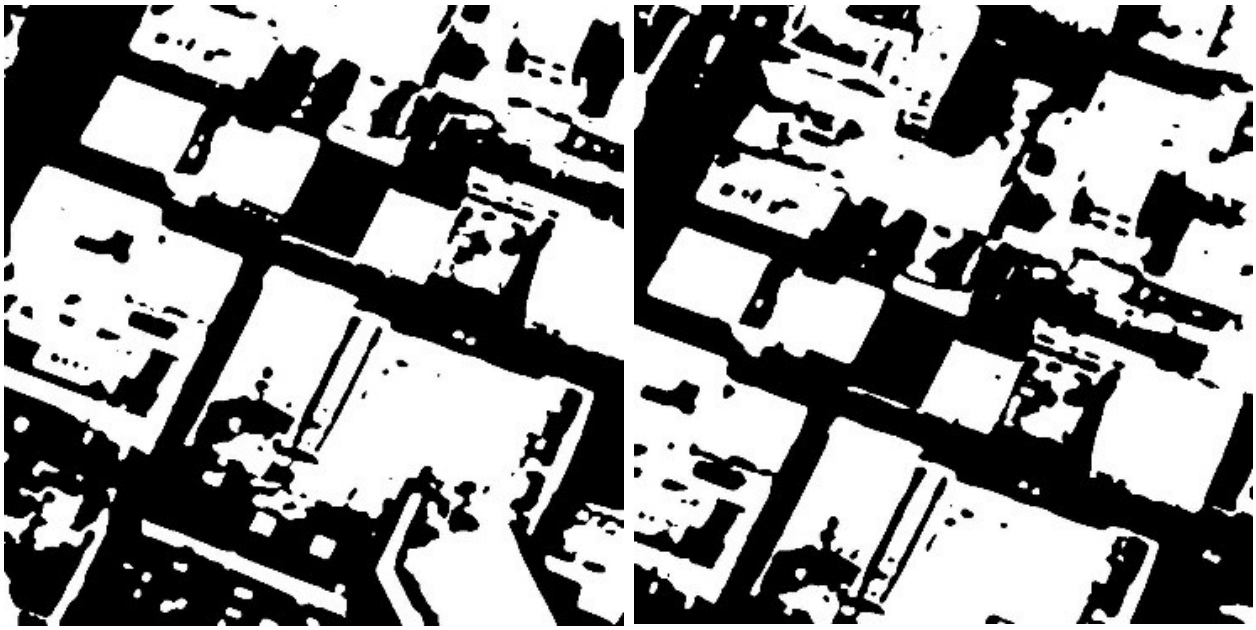


Figure 30 represents the images introduced in Figure 13 The two original images used to demonstrate BIM's stitching process after Figure 14 Illustrated BIM process used for image stitching's steps 1, 2, 3, 4 and 5, thresholding with a value of 109.

3.6.7 Shape contouring blobs comparison

One of the techniques used for blobs comparison is shape contouring. Blobs contour are approximated using The-Chin contour algorithm (Teh, 1989). It corresponds to the algorithm's (Figure 3) step 4 and is identified on Figure 14 as step 4.1. The algorithm computes each shapes and determines support regions for each points making the contour approximation, before selecting dominants points by non-maximum suppression (Jan Hosang, 2017). Resulting points are used to approximate every blobs' contours

Figure 31 illustrate the image after The-Chin contouring, border detection was applied on thresholding to make the image more understandable for the reader's eyes. On the right, the contours are applied on the thresholded image, on the right the contours are applied on the original image for comparison.

Figure 31 represents the images introduced in Figure 13 The two original images used to demonstrate BIM's stitching process after Figure 14 Illustrated BIM process used for image stitching's steps 1, 2, 3, 4, 5 and 6.a, Shape contouring using The-Chin algorithm. The right image presents shape contouring directly applied on the original image for comparison.

Shapes are compared (Rustam Paringer, 2020) Shape contours comparison is done according to the following calculation:

Equation 7 Shape contours comparison

$$I_m(A,B) = \max_{i=1..7} \frac{\left|m_i^A - m_i^B\right|}{\left|m_i^A\right|}.$$

Where $m_i^A = sign(h_i^A) \cdot \log h_i^A$ and $m_i^B = sign(h_i^B) \cdot \log h_i^B$ and h_i^A, h_i^B are the Hu moments of shape A and B , respectively. It allows to determine whether blobs present similar shapes independently from their size.

And the perimeter comparison to the following:

Equation 8 Shape perimeter comparison

$$I_p(A,B) = \left|\frac{P_A - P_B}{P_A + P_B}\right|.$$

Where P_A and P_B are the perimeter of shape A and B. Which allows to compare blobs in size.

Shape contours are considered equal if the statement $I_m < 0.15$ and I_p<0.1 is correct, thus meaning the blobs are of comparable shape and size, corresponding to the algorithm's (Figure 3) and BIM's process (Figure 14) step's 5.

Points are extrapolated from the blob's centroids c calculated by the shape's arithmetic mean according to the following formula for n points in Q.

Equation 9 Shape's arithmetic mean

$$c = \frac{1}{n}\sum_{i=1}^{n} Q_i.$$

This technique was developed in a second time as BIM struggled in detecting images presenting some specific kind of noises as described further in 3.7 Results. In general, Shape contouring allows comparison with a higher confidence and precision than convex hull. However, it is more sensitive to blobs contour alteration and therefore less noise resilient, consequently, both techniques are used in parallel.

3.6.8 Convex hull blobs comparison

The second approach selected for blobs comparison was convex hull contouring, using the, Graham scan algorithm (Ronald Graham, 1983). Convex hull consists for a set of points x to find the smallest convex polygon having in or on its boundaries the full set of points Q (Jarvis, 1973). It corresponds to the algorithm's (Figure 3) step 4 and is identified on Figure 14 as step 4.2.

Graham scan algorithm selects a point $x_0$that will necessarily be on the convex hull (such as the point having the lowest y-coordinate). Remaining points $Q_1$to Q_n are sorted ascendingly and contraclockwise by polar angle relative to Q_0. Then, points are added to the convex hull following the rule:

- If the next point $Q_i$is on the left of Q_{i-1}, $Q_i$is added to the hull H
- If the next point $Q_i$is on the right of Q_{i-1}, elements are removed from H until the first condition is true.

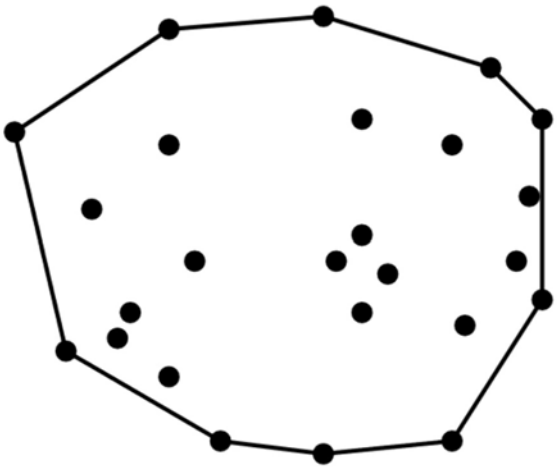


Figure 32 Convex hull H application example on a set of points Q. (Mount, 2012)

Convex hull are calculated from all blobs in the image and those answering to the conditions presented under are considered relevant as described in * (Y Donon et al, 2019).

Equation 10 Convex hull filtering

$$I_A * T_A < H_A \;\&\; H_h < I_h * T_h \;\&\; H_i < I_i * T_i.$$

For I_A, I_h, I_w the image's area, height and width and H_A, H_h, H_w the hull's area, height and width. And as threshold parameters T_A = 0.02 and T_h T_i = 0.9.

Polygons obtained from the hulls were then compared together in height, width and surface as if:

Equation 11 Convex hull comparison in area

$$\left|\frac{H_A^A - H_A^B}{H_A^A + H_A^B}\right| < T_{a\prime}.$$

Keeping a maximum difference of T_c = 0.08 and:

Equation 12 Convex hull comparison in height

$$H_h^B * T_{h\prime} < H_h^A < H_h^B * T_{h\prime\prime}.$$

Equation 13 Convex hull comparison in width

$$H_w^B * T_{w\prime} < H_w^A < H_w^B * T_{w\prime\prime}.$$

With as threshold parameters $T_{h\prime} = T_{w\prime} = 0.92$ and $T_{h\prime\prime} = T_{w\prime\prime} = 1.08$. If pairs are matched, all hulls angles coordinates are paired together, generating 3 or more pairs of points by hull in accordance to the algorithm's (Figure 3) and BIM's process (Figure 14) step number 5.

Figure 33 Shows the image after convex hull Graham scan algorithm (Graham, 1972). On the left, the hulls are superposed on the original image, on the right the hulls are superposed on the image with borders only highlighted.

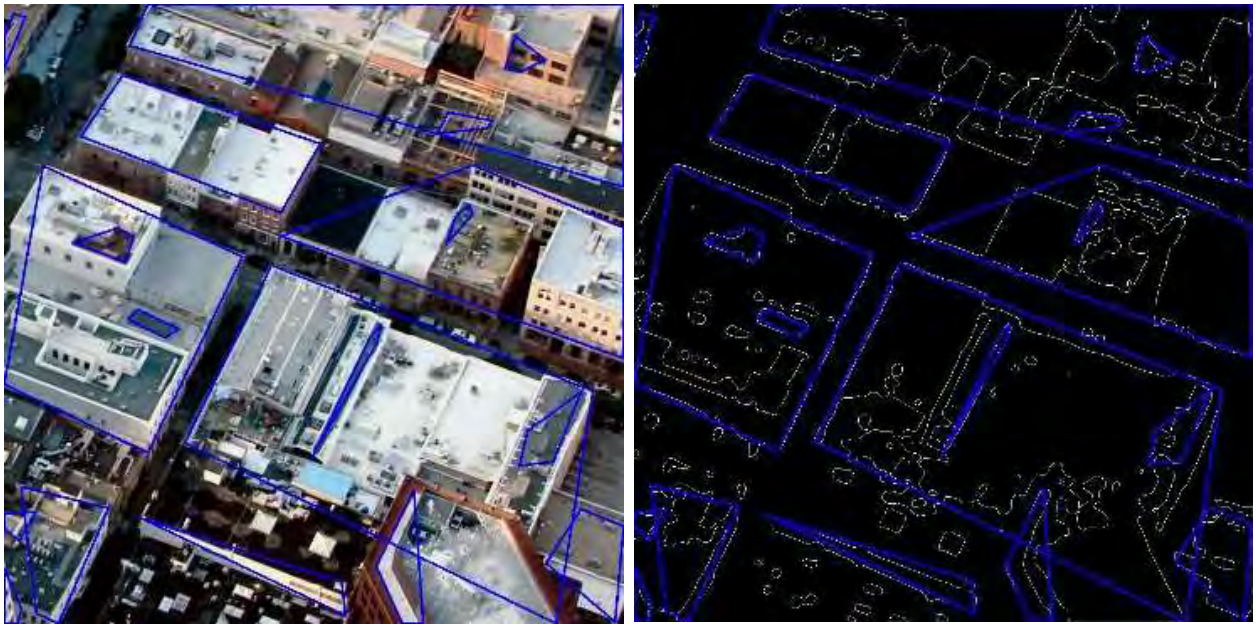


Figure 33 represents the images introduced in Figure 13after Figure 14's steps 1, 2, 3, 4, 5 and 6.b, Shape hulling using Graham scan algorithm. The right image presents shape contouring directly applied on the original image for comparison.

Convex hull is extremely resilient to noise as it is enough for the algorithm that only the shape's extremities remains relatively similar from an image to another. However, it lacks the precision offered by shape contouring.

3.6.9 Coordinates matching

Coordinates founds, by shape contouring (centroids coordinates) and convex hull application (angles coordinates) are matched together, creating a map of coordinates for the homography.

The coordinates contain outliners (highlighted in red in Figure 34) those outliners might have to be filtered out, as they have heavy consequences on some uses that can be made out of the key points, such as the homography used to compare results in this thesis. This step of filtering is done by RANSAC as described in "Results evaluation" as it is not part of the algorithm proposed.

Figure 34 represents points corelated between the pair of image introduced in 3.6.2 and at step 7 of Figure 14 Illustrated BIM process used for image stitching. Outliners are highlighted in red, those outliners perturb greatly the homography if not filtered out..

3.7 Results

As introduced in chapter 2, BIM has been compared to 3 other popular techniques in order to assess its characteristics to other techniques. All processing has been done using an Intel Core i3-8100 at 3.60GHz, which maintains the experiments in an environment comparable to portable computing capabilities. Results have been calculated using the datasets presented in * (Y Donon et al, 2019) * (Yann Donon R. P., Brightness normalization for Blurred Image Matching, 2020) (Yann Donon R. P., Parameters selection for Blurred Image Matching, 2020) * (Rustam Paringer, 2020) and * (Pierre Donon). The first set was meant to evaluate BIM's performance on regular pictures. The sample is constituted of about 225'308 images extracted from social medias presenting various exposition and subjects. The noised dataset used to compare the techniques performance with noised images contains 7872 images of different sizes and kind, including cities aerial views, paintings, landscapes and buildings. The images have been noised artificially using OpenCV using as described in 3.7.5 Stitching success rate. A third set was used, this set particularity is that it didn't contain reference images (images as they should be, if stitched perfectly). The third set was used to evaluate the technique with different parameters as well than the elements presented in 3.8.2 Stitching large amount of unordered images and 3.8.3 High confidence stitching. The sets present artificially induced

brightness differentiation between images and contains three series of drone territory mapping, for a total of 444’059 images.

The different techniques presented at the beginning of this chapter have been compared between each other, using image stitching as a reference, as described further in this chapter. The two following sub-chapters described the extra steps applied to the techniques (SURF, Harris, FREAK, BIM) to compared them to each other. The description of those steps in this thesis is made using BIM’s process as a reference.

3.7.1 RANSAC

Random Sample Consensus (RANSAC) is an iterative technique aimed to detect outliners in series of data. The algorithm approximtes results, with a probablility of success depending on the amount of iterations realized by the algorithm (Martin A. Fischler, 1980). Outliners are pairs of point significantly differing from other observations in their characteristics, in BIM, outliners are points falesely similar such as inverted angles or false positive in blob comparison (Vic Barnett, 1994).

In this application of RANSAC algorithm, an amount of itterations is done where homographies are calculated from a randomly slected sample of 4 corresponding points. Distance between where those homography transforms are then computed and classified as inliners or outliners. The operation is repeated and the itaration which produced the largest number of inliners is then selected as the best homography estimation. (Dubrofsky, 2009)

In Figure 35, the corralated points are represented after RANSAC filtering. The remaining correlations (marked in white) are all usable by the homography process.

Figure 35 represents points corelated between the pair of image introduced in 3.6.2, on Figure 34 and at step 8 of Figure 14. Here, outliners present on Figure 34 have been filtered out, resulting in correlated coordinates ready for the homography process.

3.7.2 Homography

Any number of the same planar surface-s images are related by a homography. For homogeneous coordinates (N-dimensional coordinates represented by N+1 numbers) the homography matrix H is found by the function:

Equation 14 Homography matrix

$$s_i \begin{bmatrix} x_{i\prime} \\ y_{i\prime} \\ 1 \end{bmatrix} \sim H \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}.$$

Homographical transformation consist of a mapping of point on a projective plan, concretely, it allows from the coordinates found in to build a projected imaged composed of a pair of images. (Kriegman, 2007)

The algorithm used in our context uses a minimum of four point for the projection and the entirety of the points correlated by BIM, after RANSAC filtering. Four coordinates are necessary to represent the image's coordinate in x, y their inclination and orientation.

Figure 36 illustrates the result of image stitching after BIM process of feature image processing and feature recognition.

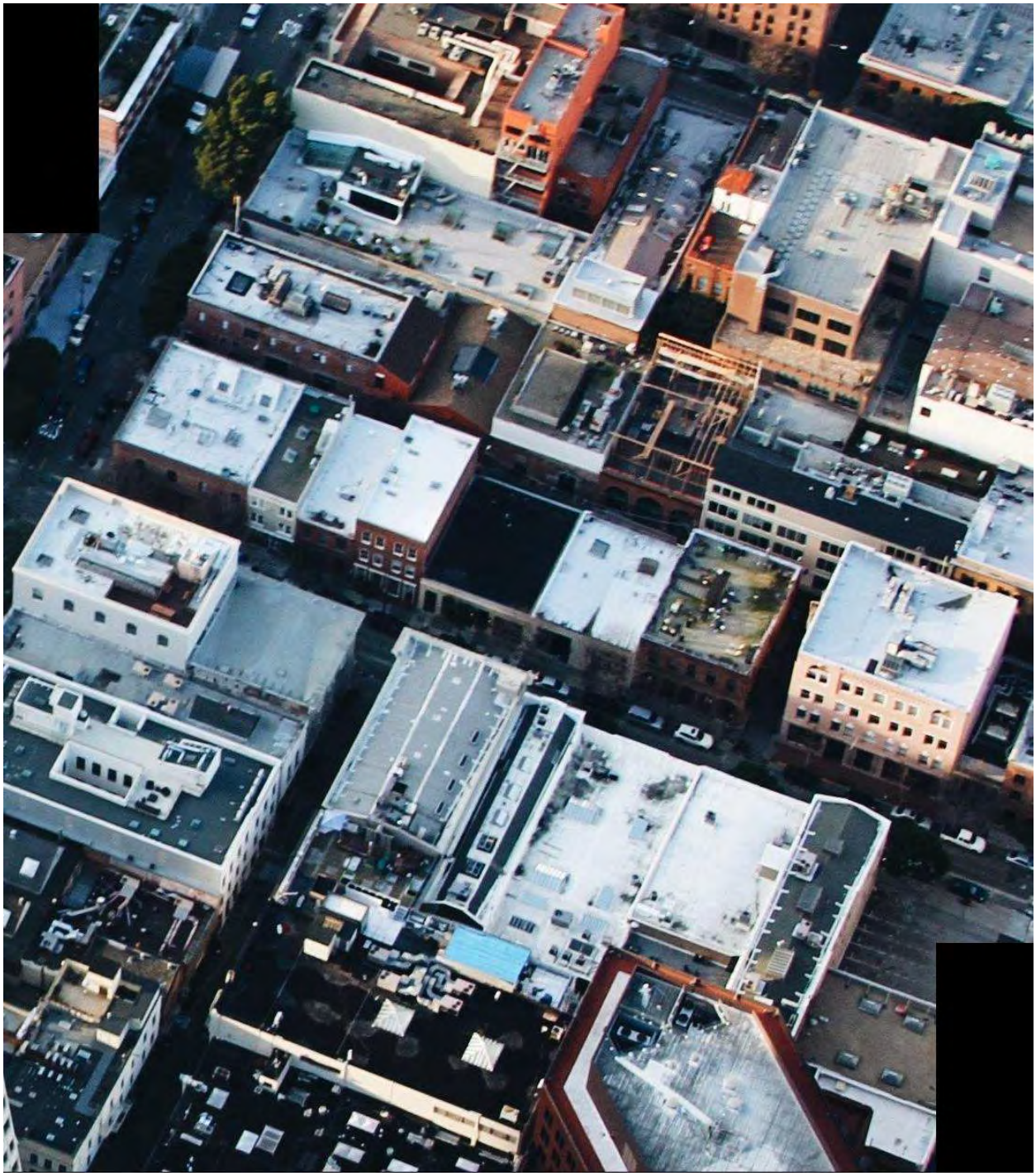


Figure 36 illustrates the image represented as example in this chapter and on Figure 14 Illustrated BIM process used for image stitching after the complete process of feature selection and image stitching.

3.7.3 Results evaluation

Results presented in this chapter have been compared using an implementation (Krarup, 2018) of the Bhattacharyya distance histogram algorithm (Bhattacharyya, 1943), the implementation involve an image transformation to grayscale and a resizing to 16x16 pixels. Bhattacharyya distance d is calculated between two images' histograms H and H' with H_i and H'_i the i^{th} interval of H and H' dependant on all pixels as with the number of interval N = 16^2 as:

Equation 15 Bhattacharyya distance

$$d = \sqrt{1 - \sum_{i=1}^{N} \sqrt{H(H_i)H(H'_i)}}.$$

The images taken to compare the performance are issued from different sets * (Y Donon et al, 2019) containing original images that have been cut according to random patterns and on which noise has been applied. This signifies that the sets contain both image fragment and final, "perfect" image results. In this work, it was decided based on estimation that only images with a Bhattacharyya difference of 2% or less would be considered similar as represented on Figure 37. Images presenting a Bhattacharyya difference of 2% are presented on Figure 38, the left image is the result of a lack in precision in the homography matrix, the right image is the typical result of a stitching attempts with a homography matric corrupted by one or more pair of points as presented on Figure 35. Images are presented with the 16x16 array extrapolated from the transformed image used for the Bhattacharyya difference as described earlier in this chapter. Every square contains a pixel's Bhattacharyya difference, the image's Bhattacharyya difference is calculated from the average of each pixel difference.

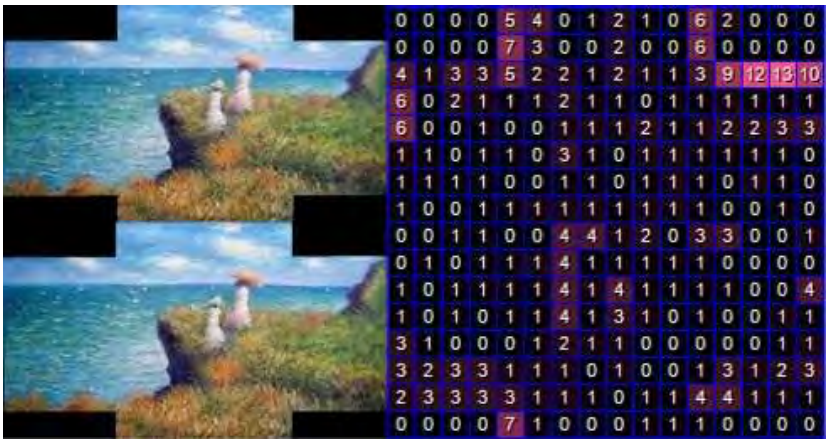


Figure 37 Comparison between two successfully matched pictures, with a Bhattacharyya difference index of 1.5%. Between the original mage (top) and stitched image (bottom), no difference appears to the human eye.

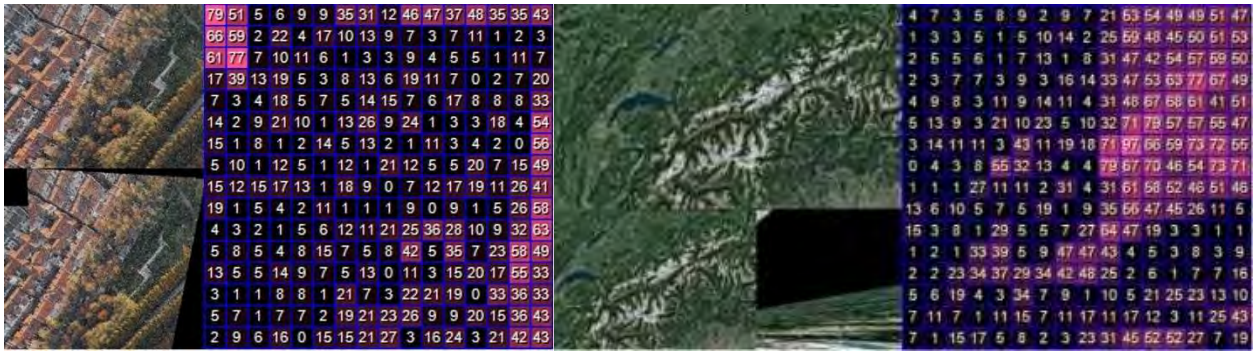


Figure 38 Comparison between two unsuccessfully matched pictures, with a Bhattacharyya difference index of 19.6% (left) and 36.8%. (right). Expected result appears on top and stitching attempts on the bottom.

3.7.4 Processing time

Image processing time, which represents the full process described in 3.6.1 Process presentation is in shorter than other implementations on tested values. The difference in processing time is mainly explained by the complexity of researched features, which depends on the technique and the number of features found, which greatly complexifies the operation of filtering and homography matrix calculation as highlighted in Figure 39.

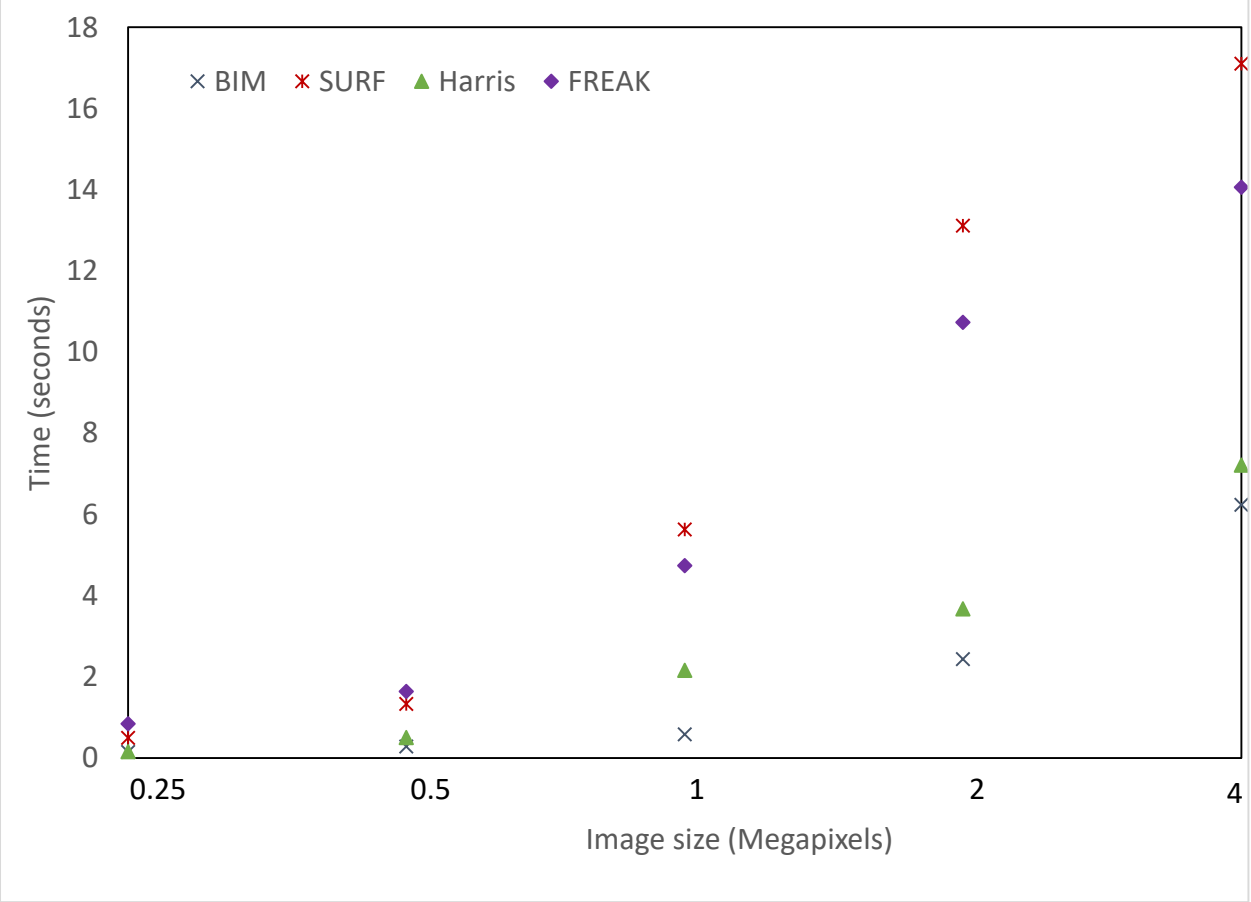


Figure 39 Computation time comparison between the different techniques. In this figure BIM stitching takes significantly less time than with other techniques, independently from image sizes.

Figure 39 shows the feature calculation and image stitching processing time on 5 different datasets, containing series of images of 0.25, 0.5, 1, 2 and 4 megapixels. Only successfully stitched images, as defined in 3.7.3 Results evaluation, were selected. On average, BIM is 71% faster than other techniques and 34% faster than Harris, its closest comparison in terms of speed.

BIM speed is due to two factors, first being that after the image analysis, the feature identification process is more basic than in other techniques, borders are fewer than on other images due to the Gaussian blurring application and the image itself is thresholded, making the

borders simple to identify. However, the main factor of speed at this stage is the amount of points detected. As shown on Figure 40, BIM finds on average significantly less points than other techniques, namely 49 times less than the technique finding the highest number of correlations (FREAK) and 7 times than the second technique in this category (SURF).

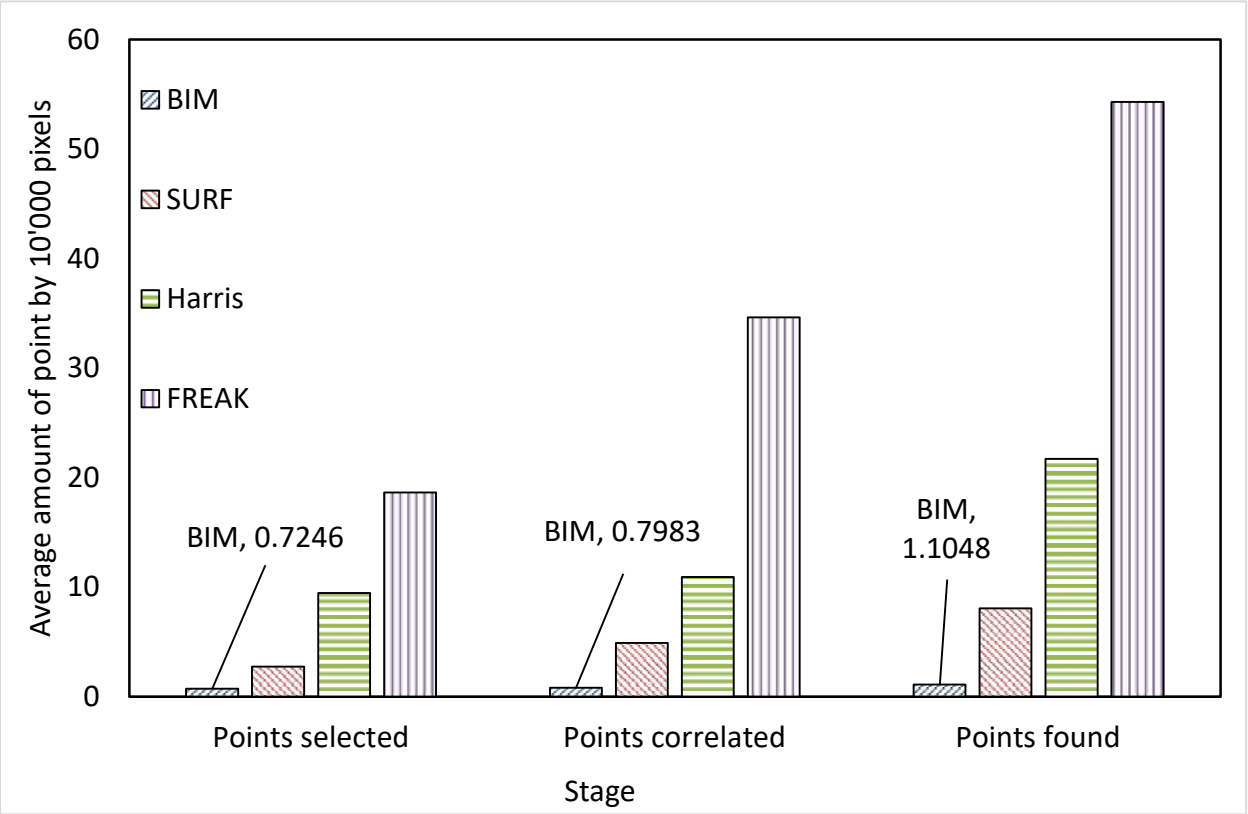


Figure 40 Average points found by technique on 10k pixels images. This figure highlights how fewer points are selected by BIM compared to other techniques.

This feature of BIM is tightly attached to the technique's concept of finding large areas of interest on images. It is possible to augment in specific cases the number of features detected as described further in 3.8.3 High confidence stitching. for some specific uses. However, for most uses, the small amount of points is a useful feature, it is an important element of the short processing time demonstrated by BIM as in Figure 39 but this features fain in importance for the problem of large unordered datasets as described further in 3.8.2 Stitching large amount of unordered images. For such datasets, the feature detection time on images grows linearly for each added image, however, the feature comparison time grows exponentially, and less features heavily impact on reducing calculation time in such situation.

If BIM detects a small number of features compared to its counterparts, it is also the technique detecting features with the highest quality. A features quality is defined by the

number of features usable for the homography matrix estimation out of the original pool of features detected.

Figure 41 Shows the percentage of points kept by technique after the matching stage and after the RANSAC stage, a hundred percent being the total amount of features detected. The bars with a light background represent the number of features matched, every feature finding a pair from an image to the other. A higher percentage means that detected features are more significant, thus reducing the noise induced by other features. This noise is relevant as it can be the source of false positive in matching. BIM is the technique having the most points matched after FREAK and SURF. The bars with a dark background represent the proportion

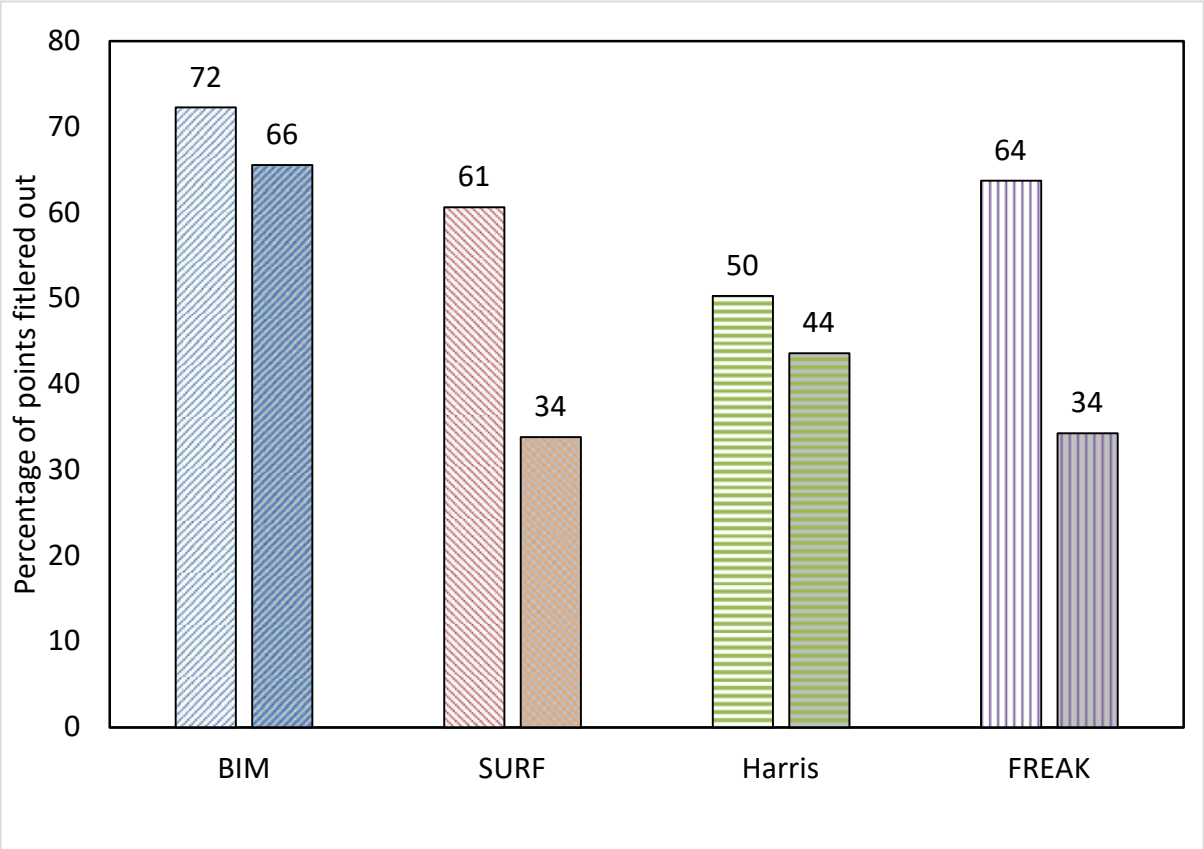


Figure 41 Percentage of points kept after the matching and filtering operations. This figure shows BIM's higher ratio of points usable for the homography. A higher ration means a higher point's quality. Bars with light background represents the proportion of features matched with a pair, Bars with dark backgrounds represents the proportion of features left after RANSAC filtering.

of images left after RANSAC filtering. A higher percentage means less false positive in features comparison (matching), which diminishes the chances of the RANSAC algorithm estimating a corrupted homography matrix. BIM and Harris present on average 6% of false positive as evaluated by the RANSAC algorithm, by far inferior to SURF (27%) and FREAK (30%). As such, BIM is the only technique able to use most detected points for the final homography with 66% against Harris 44%, and 34% for FREAK and SURF.

Such feature is extremely important for the techniques used for the stitching of large amount of images (3.8.2) not only because of the reduced amount of features but also as the algorithm used for stitching depends heavily on the probability that detected points are relevant.

3.7.5 Stitching success rate

BIM was compared to SURF, Harris and FREAK in terms of success rate on two sets of images, one containing image purposely noised to different degrees and one containing regular images. Success as defined in 3.7.3 were measured and compared on both sets, as illustrated on Figure 42. The figure shows on bars with light background success in percent on the regular dataset and on bars with a dark background success rate on the noised dataset.

As shown on Figure 42 (light bars), BIM registers excellent success rate (93.8%), just under Harris (94.2%), followed by SURF (59%) and FREAK (49%) when it comes to stitch non-noised images. It seems important as this points to underline that this statement is no case an evaluation of the general quality of a technique but only a comparison to BIM in a restricted framework. It is however enough to assess BIM as usable and to some extent competitive

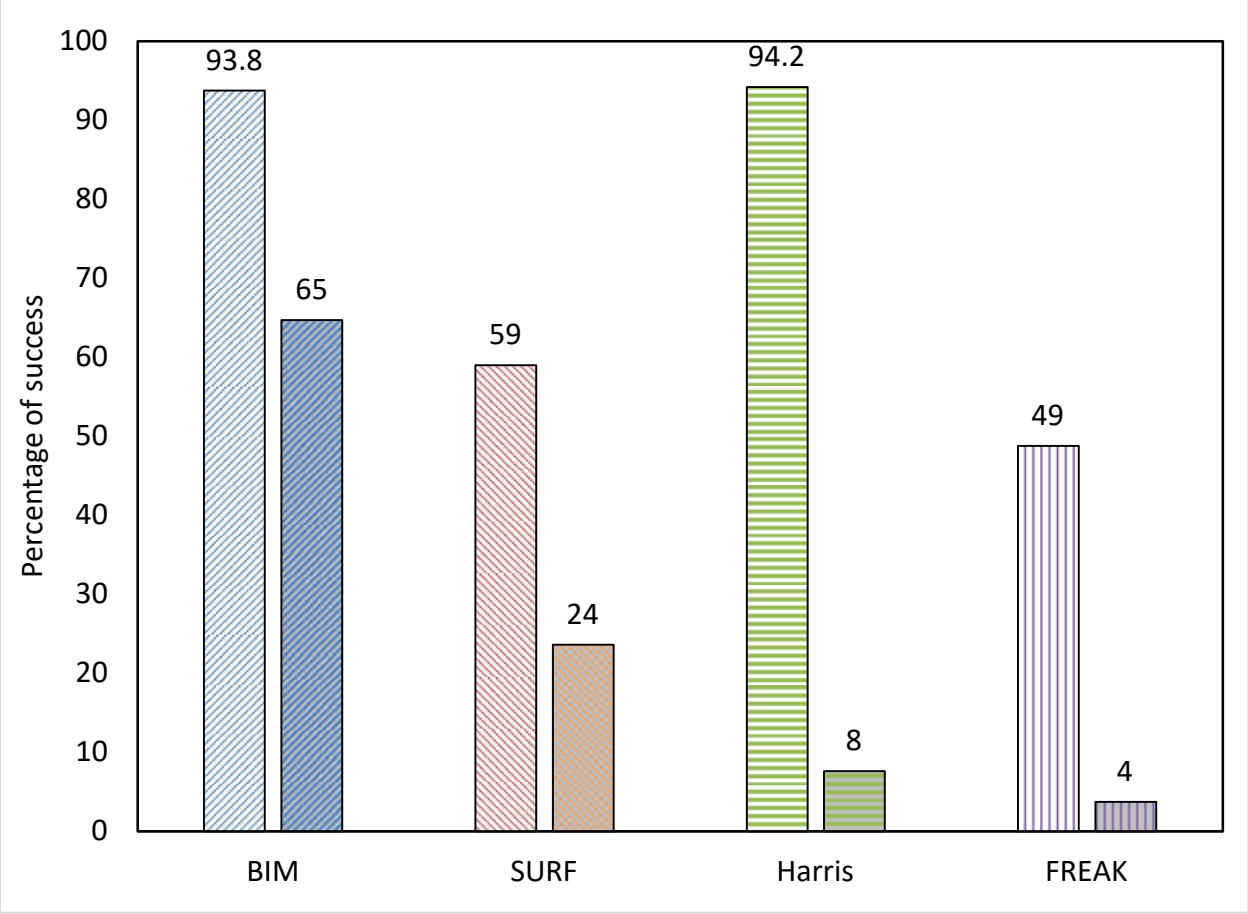


Figure 42 Techniques' success rate comparison in percent, on both sets. In this figure BIM presents the second highest performance on the regular set and the highest on the blurred set.

among its pairs to deal with cases of feature detection and comparison on non-noised images. Thus, making the technique polyvalent.

Figure 42 also shows (dark bars) success rate on purposely noised images set. As presented initially in 1 Aim, previous techniques application didn't exceed 24% of success rate (SURF) and was followed by Harris (8%) and FREAK (4%), thus underlining a need for a robust technique. On the same dataset, BIM registers a success rate of 65%.

The purposely noised dataset is of images on which different kind of alteration have been overlaid, noises have been selected in order to imitate commonly encountered noised on images. Some noise are the combinations of two different noise filters. Noise filters are:

- Figure 48.a Gaussian blurring using a normalized box filter [17], with a kernel size of (x, x), x comprised between 5 and 200 incremented by steps of 5. Typically found in out of focus captures or captures in movement. The noise's probability density function p of a Gaussian random variable g is defined as follow (Ribeiro, 2004):

Equation 16 Gaussian random variable's probability function

$$p_G(g) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(g-\mu)^2}{2\sigma^2}}.$$

- Figure 48.b Artificial perspective with a deformity from the point of origin up to 1/4 of the image's size. This kind of noise is characteristic from images taken from different angles or moving objects. The function calculates the perspective transformation as for the matrix's map m (OpenCV, 2020):

Equation 17 Perspective transformation matrix

$$\begin{bmatrix} t_i x_{i'} \\ t_i y_{i'} \\ t_i \end{bmatrix} = m \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}.$$

Where

Equation 18 Perspective transformation matrix parameters

$$dist(i) = (x_{i'}, y_{i'}), src\ (i) = (x_i, y_i), i = 0,1,2,3.$$

- Figure 48.c A multiplicative noise (speckle noise) depending on picture's height, width and depth, all with an intensity between 500 and 20'000. Such

noise exists in in active radars, Synthetic Aperture Radars (SAR, used to recreate three dimensional objects out of two dimensional images) (Jong-Sen Lee, 2009) and medical ultrasound (Forouzanfar, 2007). The noise density d is modelled as:

Equation 19 Single-Look Multidimensional Speckle Noise Model Hermitian product of two SAR images

$$S_i S_j^* = \psi \overline{z}_n n_m N_c \, exp(j\phi_x) + \psi\left(|\rho| - N_{c\overline{z}_n}\right) exp(j\phi_x) + \psi(n_{ar} + jn_{ai}) \quad (7).$$

Where $S_i S_j$, for i, j=1,2,...,m, are the Hermitian product of two elements of the zero mean vector $\mathbf{k} = [S_1, S_2, \ldots, S_m]^{\mathrm{T}}$ and $\psi, \overline{z}_n, N_c, \phi_x$ are parameters depending on the Same Hermitian product. ψ|ρ|exp(jϕx) is the term representing the useful signal component containing the given speckle noise (Carlos Lopez-Martinez, 2007).

- Figure 48.d A filter presenting fixed defaults as presented in Figure 4, adding a white layer and lenses partial obstructions to represent the picture taken in a situation of extreme cloudiness with a damaged objective. The filter has been overlaid with an intensity varying from 2.5% to 97.5%, by steps of 2.5%. Such noise it typical of captures issued from drones. It corresponds to a multiplied blend mode the formula B is modelled for the backdrop layer C_b and the source layer C_s as:

Equation 20 Images multiplied blend mode

$$B(C_b, C_s) = C_b, * C_s.$$

- Figure 48.e Salt-and-pepper noise with a probability of the noise varying linearly from 0.01 to 0.4 α. This type of noise is often the result of electromagnetic interference (Sue, 1981). The model for an image transformation from its original $f(x, y)$ to its noised $q(x, y)$, considering MIN and MAX the limit values of a colour channel in the image, can be represented as (Boncelet, 2009):

Equation 21 From original to Salt-and-pepper noised image transformation model

$$\mathrm{PR}[\mathrm{q} = \mathrm{f}] = 1 - \alpha.$$
$$\Pr[\mathrm{q} = \mathrm{MAX}] = \alpha/2.$$
$$\Pr[\mathrm{q} = \mathrm{MIN}] = \alpha/2.$$

- Filtering and blurring as mentioned on Figure 48 is the combination of the filter noise presented above and Gaussian blurring. Both noises are applied with the same intensities and increment than presented above and following the same models.
- Perspective and blurring as mentioned on Figure 48 is the combination of the artificial perspective noise presented above and Gaussian blurring. Both noises are applied with the same intensities and increment than presented above and following the same models.

Figure 48.a to Figure 48.e, right frame are all images on which BIM could find significant enough features to ensure a matching with a pair of the same noise intensity.

Figure 48 shows the performances of BIM when confronted to different kind of noises, showing the capacities of the technique in dealing with different kind of common noises as described above.

BIM shows excellent performances when matching images affected by heavy Gaussian blur, with 93% of successful matches on the dataset, ahead of SURF (57%), FREAK (6%) and Harris (5%). This success comes without surprise as BIM functioning is based on the use of Gaussian blurring. Depreciated performances in this category is explained by the heavy impact blurring has on lines and corners as used by FREAK and Harris.

On images mimicking the effect of clouds and partial lenses obstruction, BIM shows a success rate of 78% ahead of SURF (37%), FREAK (6%) and Harris (0%). Harris tend to catch details due to lenses obstruction as features, creating false positive. FREAK also suffers from that bias but is slightly less affected as the technique selects on average more points, making filtering out false positive more likely. It is however necessary to highlight that using the convex hull features matching technique only as described 3.6.8, BIM scored 28% of success rate, again, the implementation of blob contouring was partially motivated by this lack of performances. The shape contouring feature detection is therefore to be considered extremely performant to handle this kind of noise, improving successful stitching in this category of noise by 50%.

Filtered and blurred images present remarkably similar results from the two noises previously mentioned, as it is the combination of both. BIM succeeds in 72% of cases, in front of SURF (36%), FREAK (7%) and Harris (0%). The explanations of Harris lack of performances in this precise case are the same than above, the results showed by SURF and

FREAK, ±1% from previous results can be attributed to a delta in the experimentation process. The 6% delta showed by BIM is significant enough to observe that the technique reaches its tolerance limit earlier than on single noises.

Noised induced by perspective is treated with relatively good results by Harris, with a success rate of 31%, behind BIM's 82%. It is however also relevant to note that using the convex hull features matching technique only as described in 3.6.8, BIM scored under 25% of success rate, again, the implementation of blob contouring was partially motivated by this lack of performances. The shape contouring feature detection is therefore to be considered extremely performant to handle this kind of noise, improving successful stitching in this category of noise by 57%. SURF shows performances lower than its average (5%) and FREAK (4%)

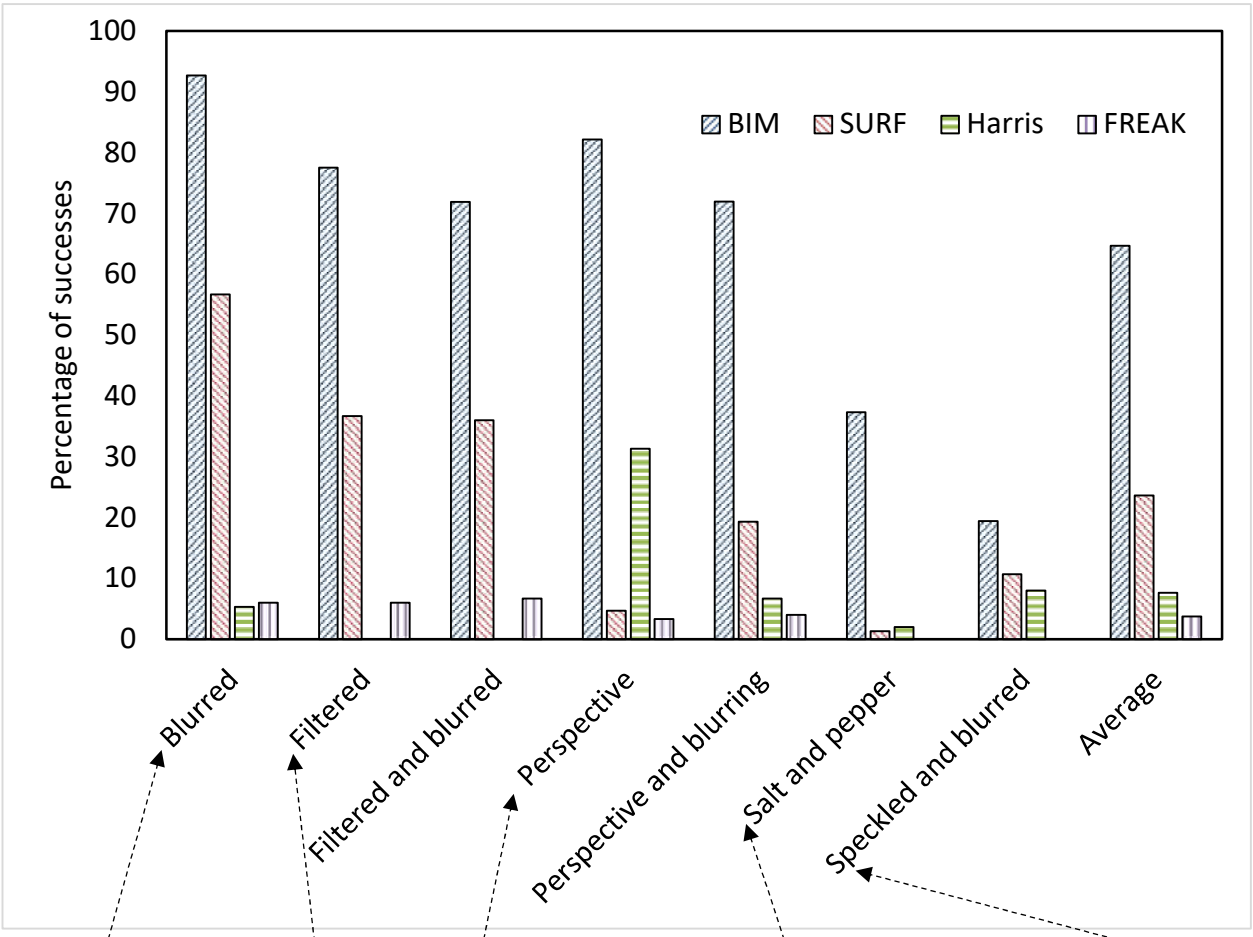
Combined perspective and gaussian blurring noise considerable diminish the performances of Harris, bringing the technique's success rate down to 7%, where the blurring operation gives better performances to SURF (19%), keeping FREAK at 4%.

Salt and pepper noise showed itself challenging for all techniques, with BIM scoring 37%, SURF 1%, Harris 2% and FREAK 0%. Highlighting a need for a technique able to compensate the lack of existing solutions. The lack of performance of existing techniques is easily explained by the fact Salt and pepper noise tends to hinder corners and edges and modify greatly colours distribution used by existing algorithms.

Multiplicative noise also, for the same reasons than Salt and pepper noise, was showed as challenging for all techniques, with a success rate of 19% for BIM, 11% for SURF, 8% for Harris and 0% for FREAK.

On average on noised dataset, as shown earlier, BIM's success rate is of 65%, SURF 24%, Harris 8% and FREAK 4%.

The following pages (Figure 49 to Figure 55) presents the different categories of noise displayed on Figure 48 with increasing intensity. Showing a progressive success reduction with the noise augmentation. All graphs are presented with a sample of image taken at the extreme point where matching between the two images is still successful according to the experiment presented.



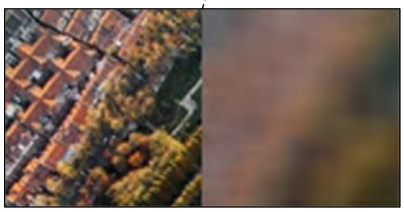


Figure 48.a Comparison between a picture before (left) and after a blurring operation with a kernel size of 200 by 200 (right).

Figure 48.b Comparison between a picture before (left) and after a perspective noise application with an inclination of 200px (right).

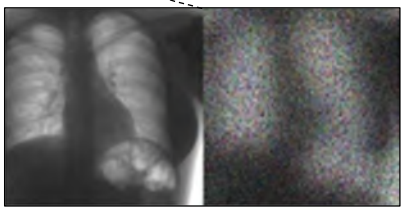


Figure 48.c Comparison between a picture before (left) and after a multiplicative and blurring noise operation with a kernel size of 150 by 150 and an intensity of 7'500 (right).

Figure 48.d Comparison between a picture before (left) and after a filter application with an intensity of 37.5% (right).

Figure 48.e Comparison between a picture before (left) and after a salt-and-pepper noise application with a probability of 0.4 (right).

Figure 48 Success rate comparison between techniques on given noises. BIM shows superior performances in all categories of noise tested. With an average success rate of 65%, 41% higher than the second-best performing technique (SURF).

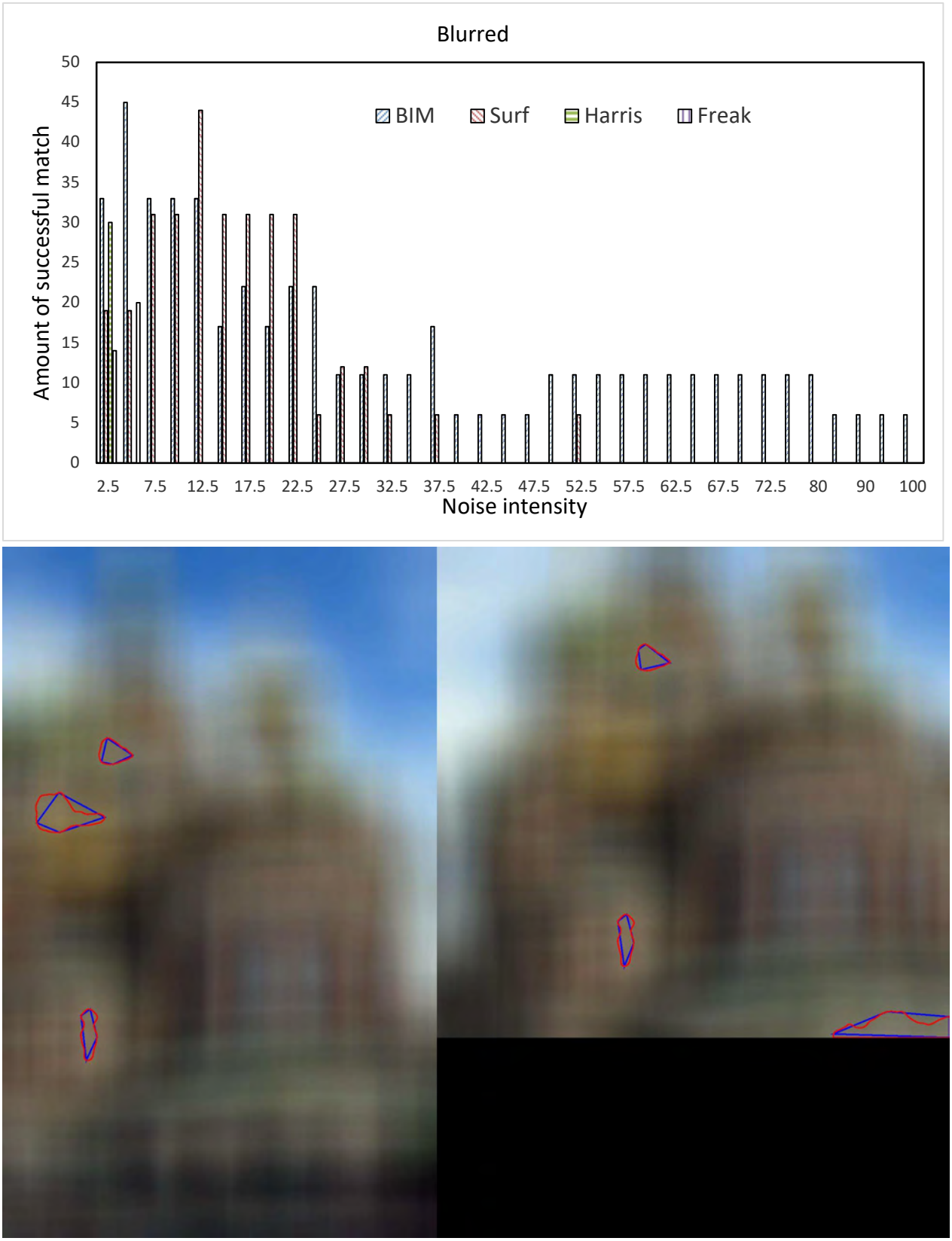
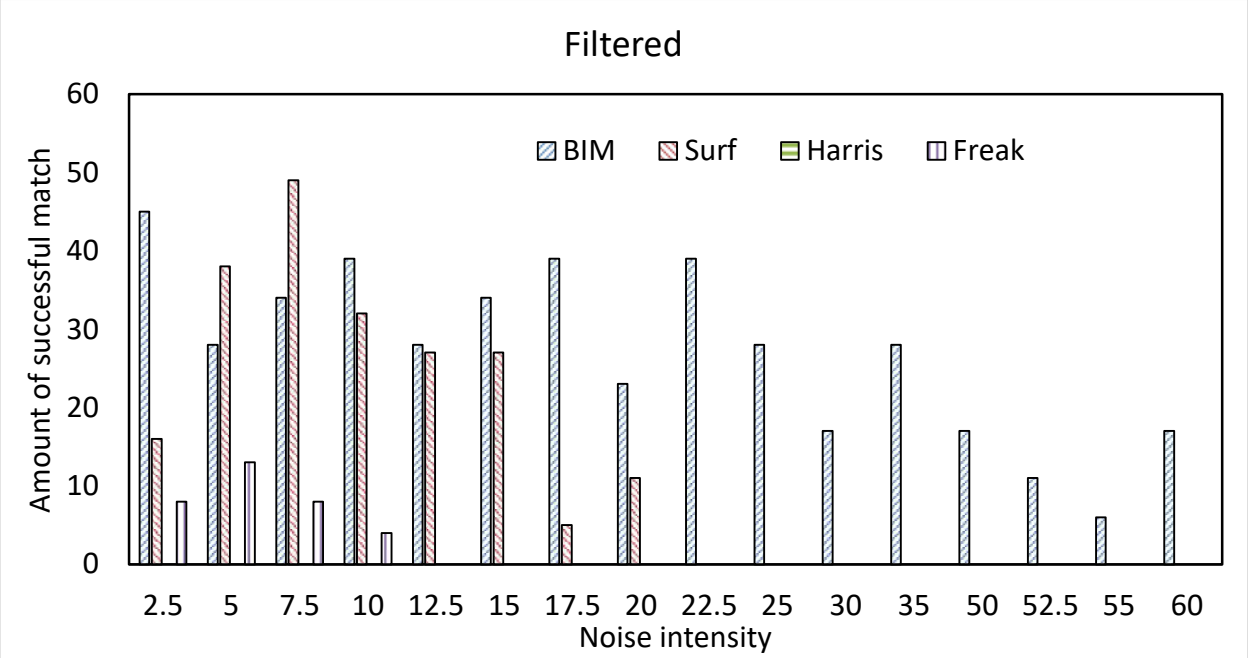


Figure 49 Amount of image successfully stitched depending on blur intensity, with a sample of image at the maximum noise tolerance.



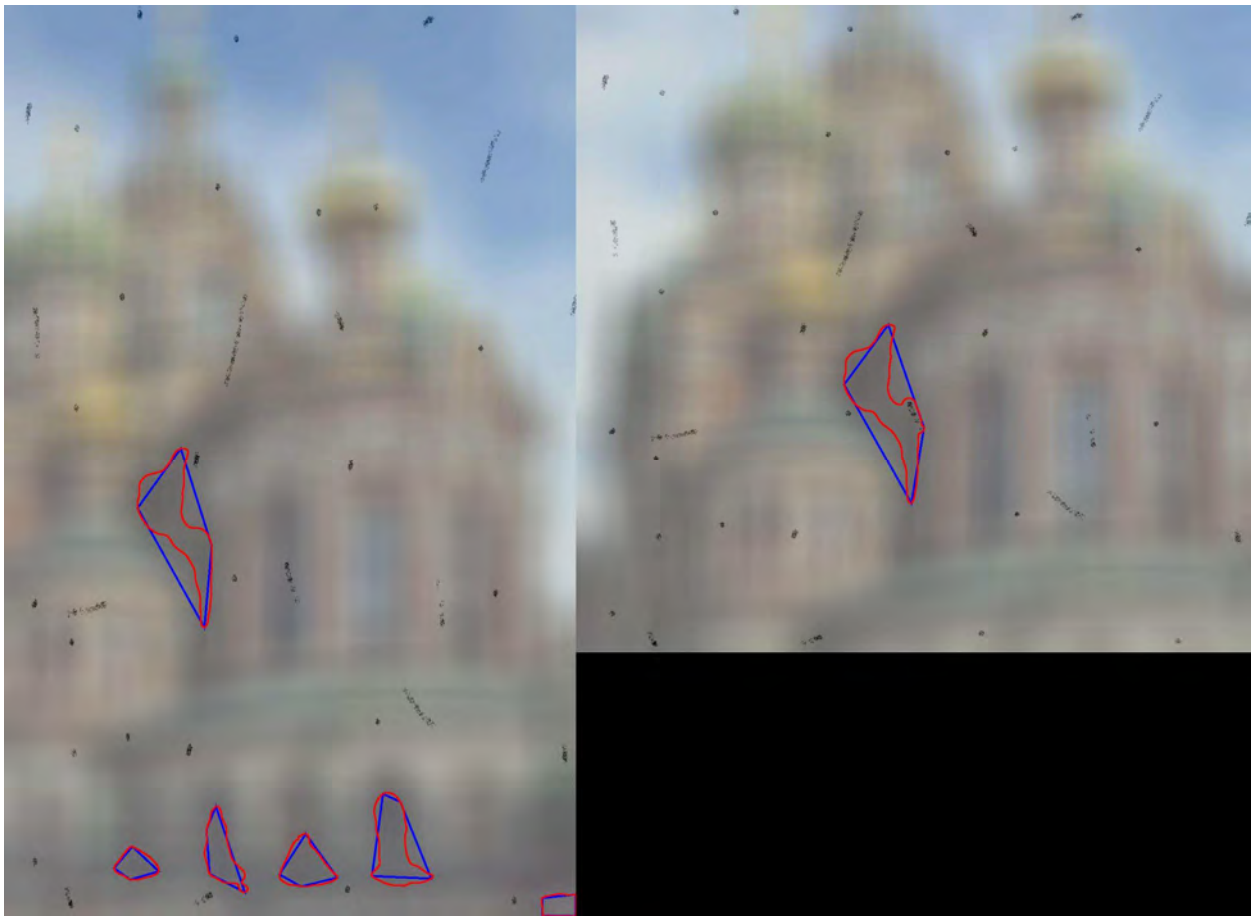


Figure 50 Amount of image successfully stitched depending on the filter intensity, with a sample of image at the maximum noise tolerance.

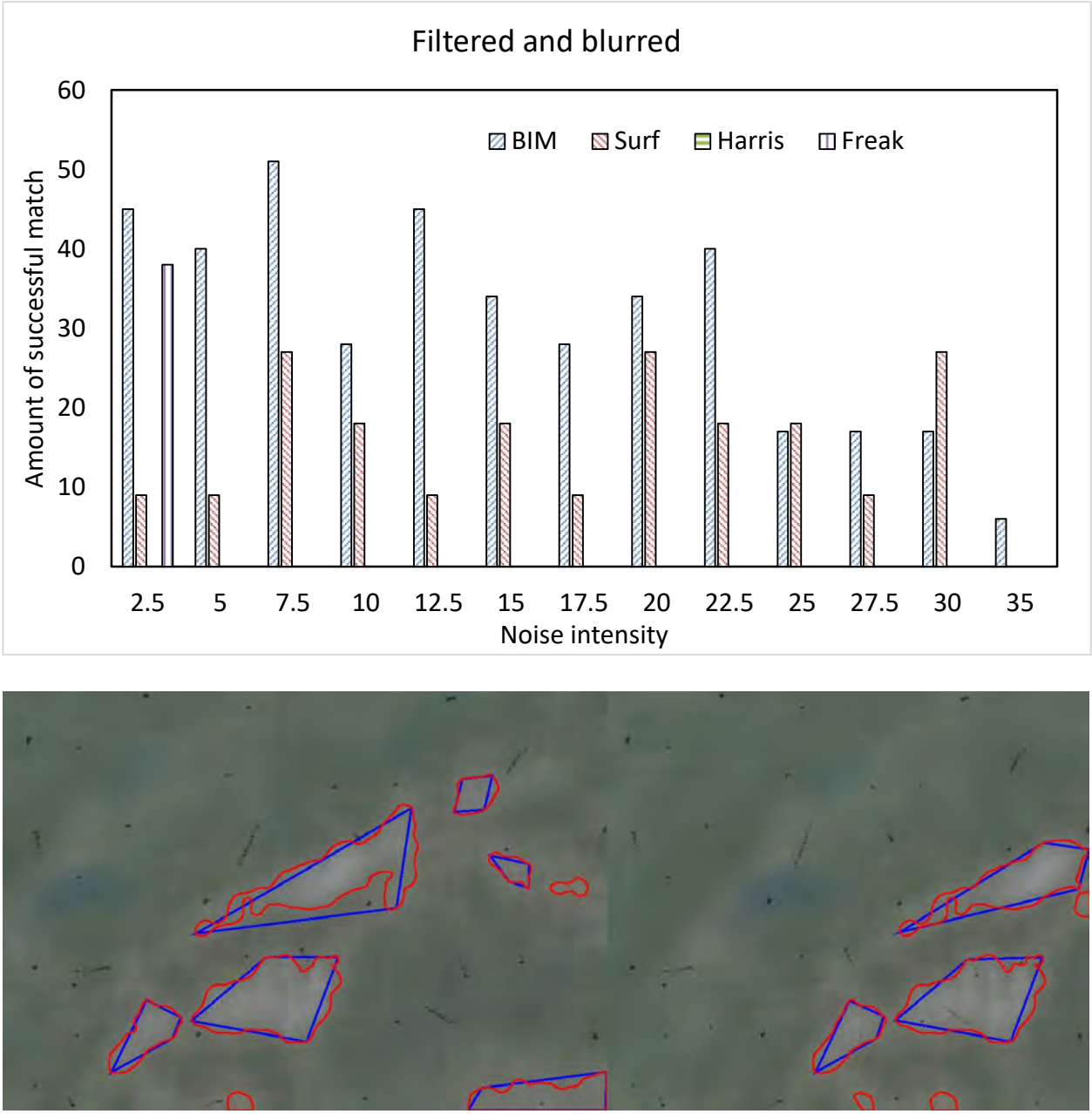
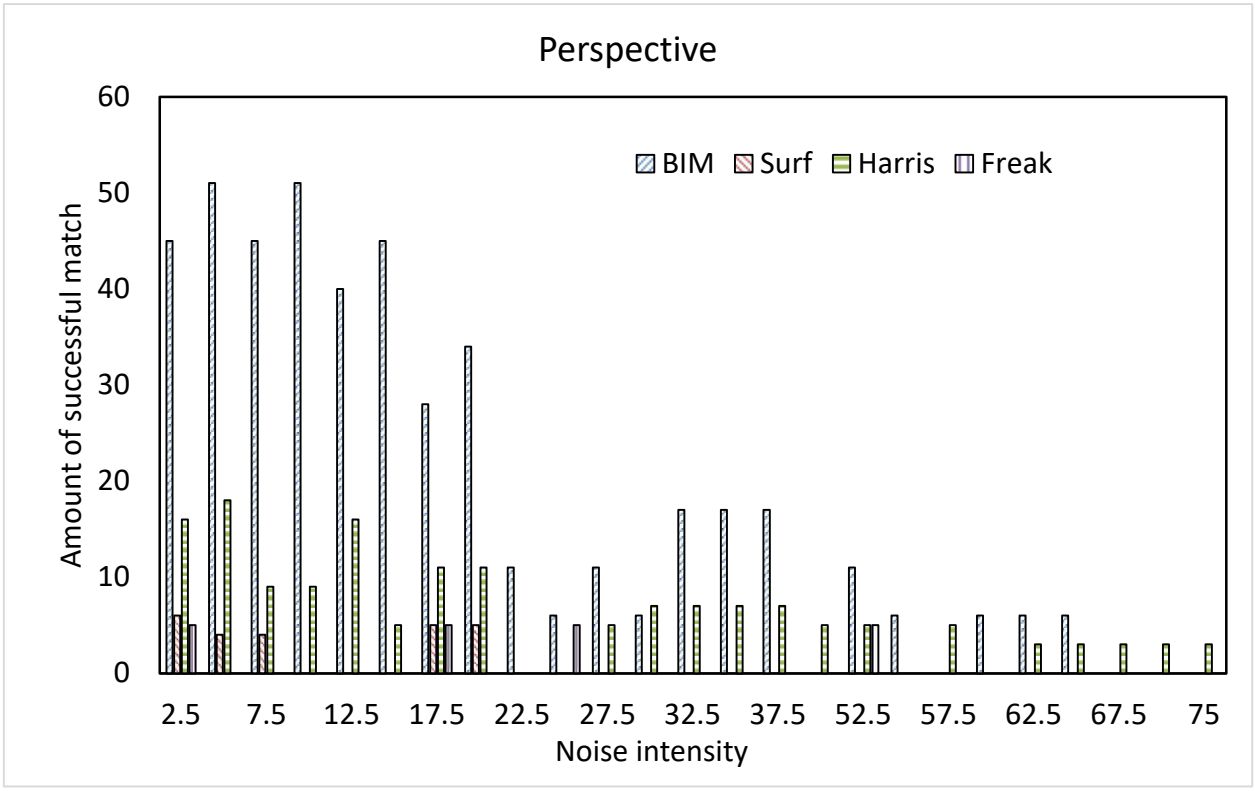


Figure 51 Amount of image successfully stitched depending on filter and blur intensity, with a sample of image at the maximum noise tolerance.



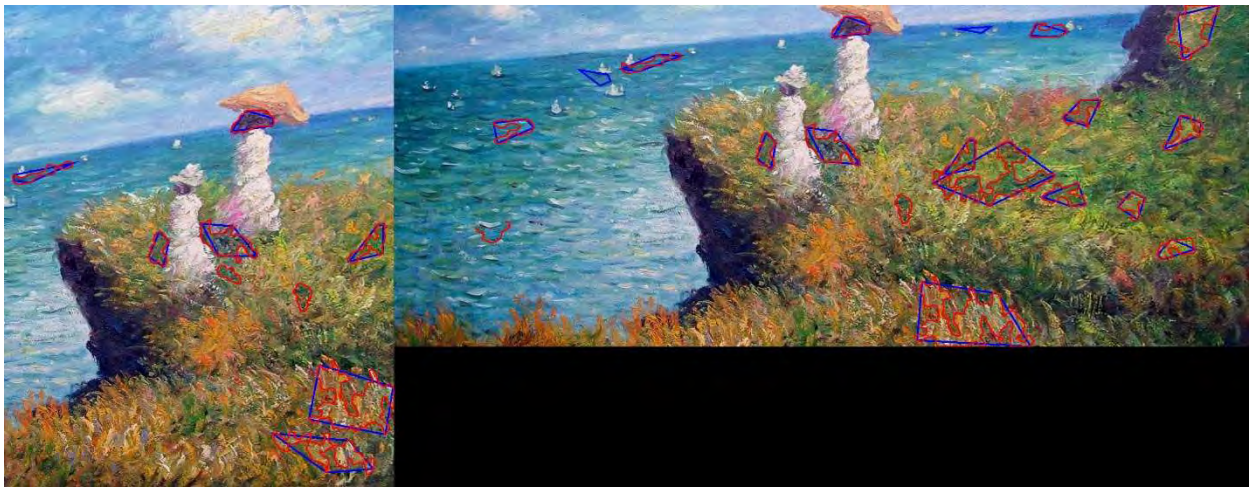


Figure 52 Amount of image successfully stitched depending on perspective intensity, with a sample of image at the maximum noise tolerance.

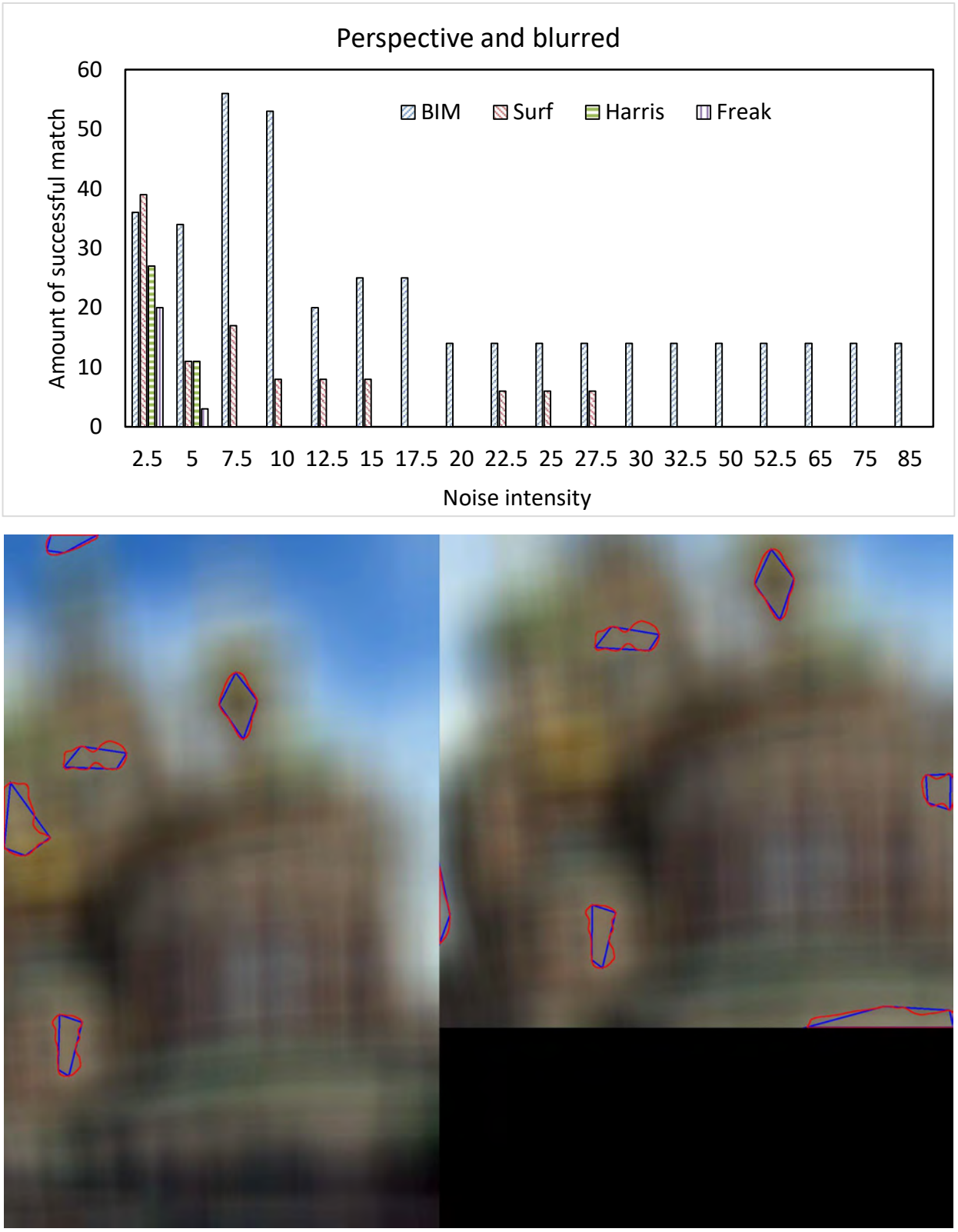
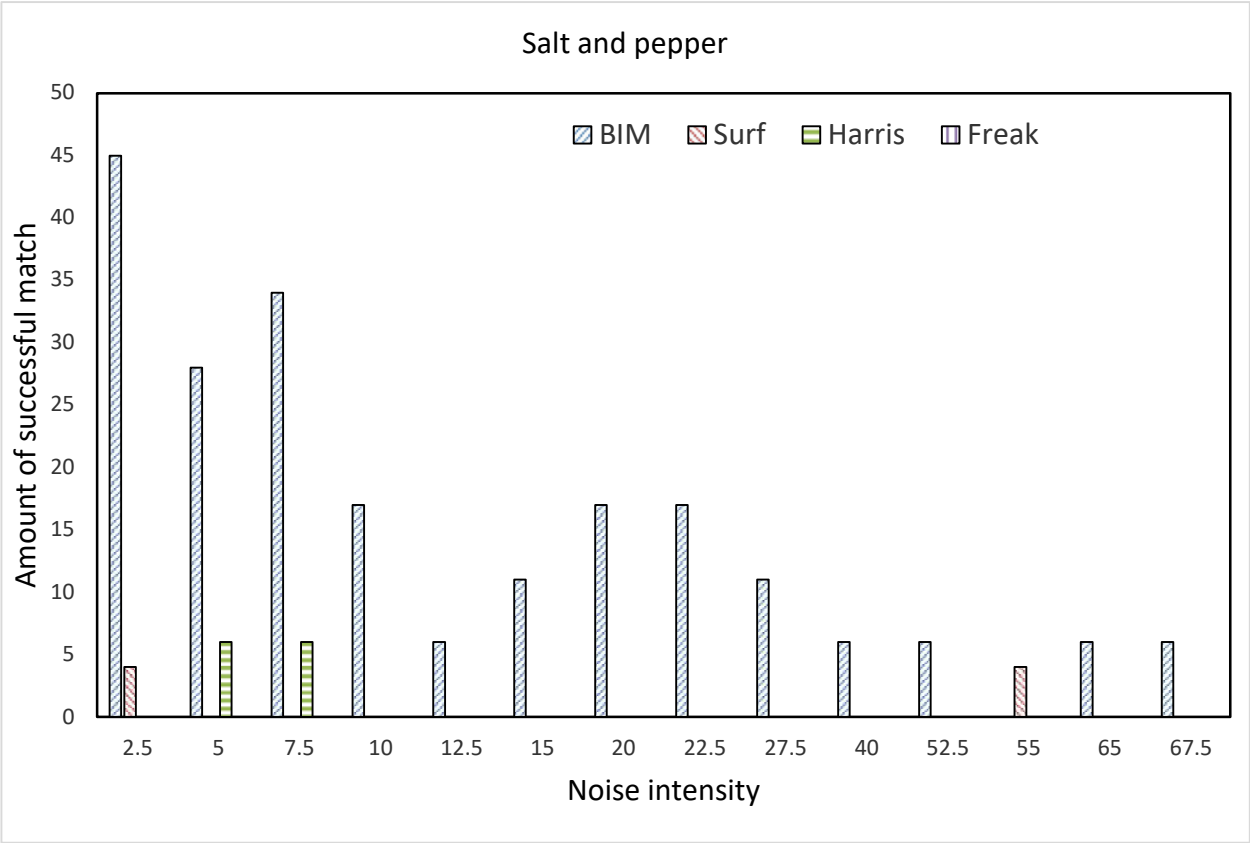


Figure 53 Amount of image successfully stitched depending on perspective and blur intensity, with a sample of image at the maximum noise tolerance.



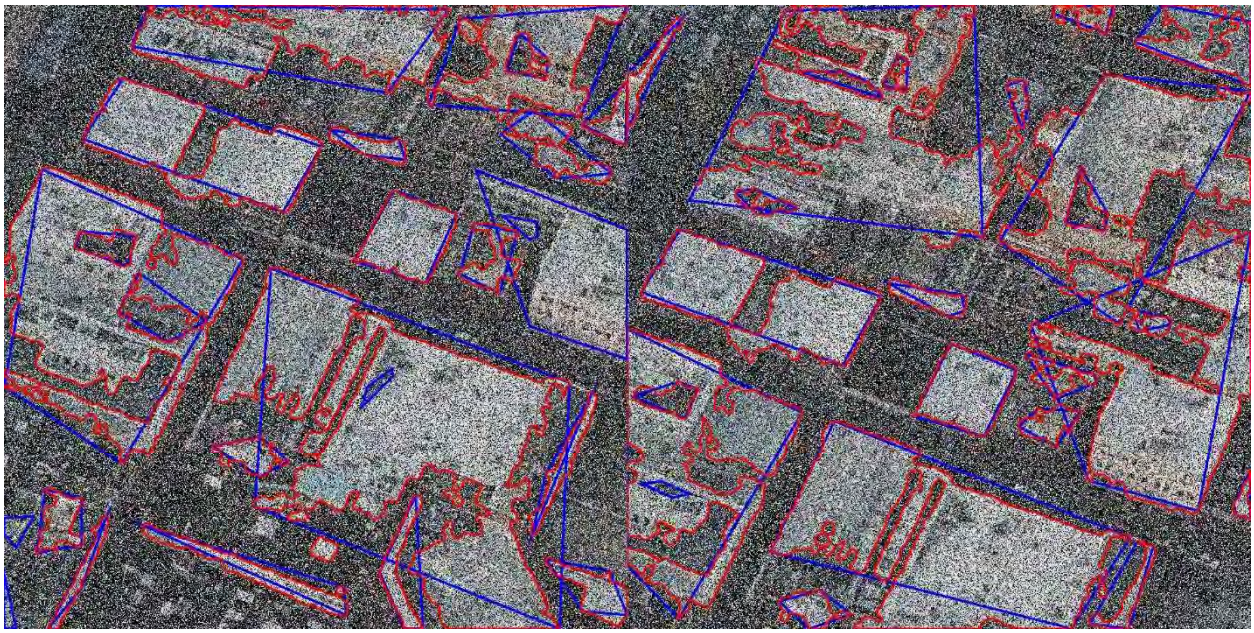
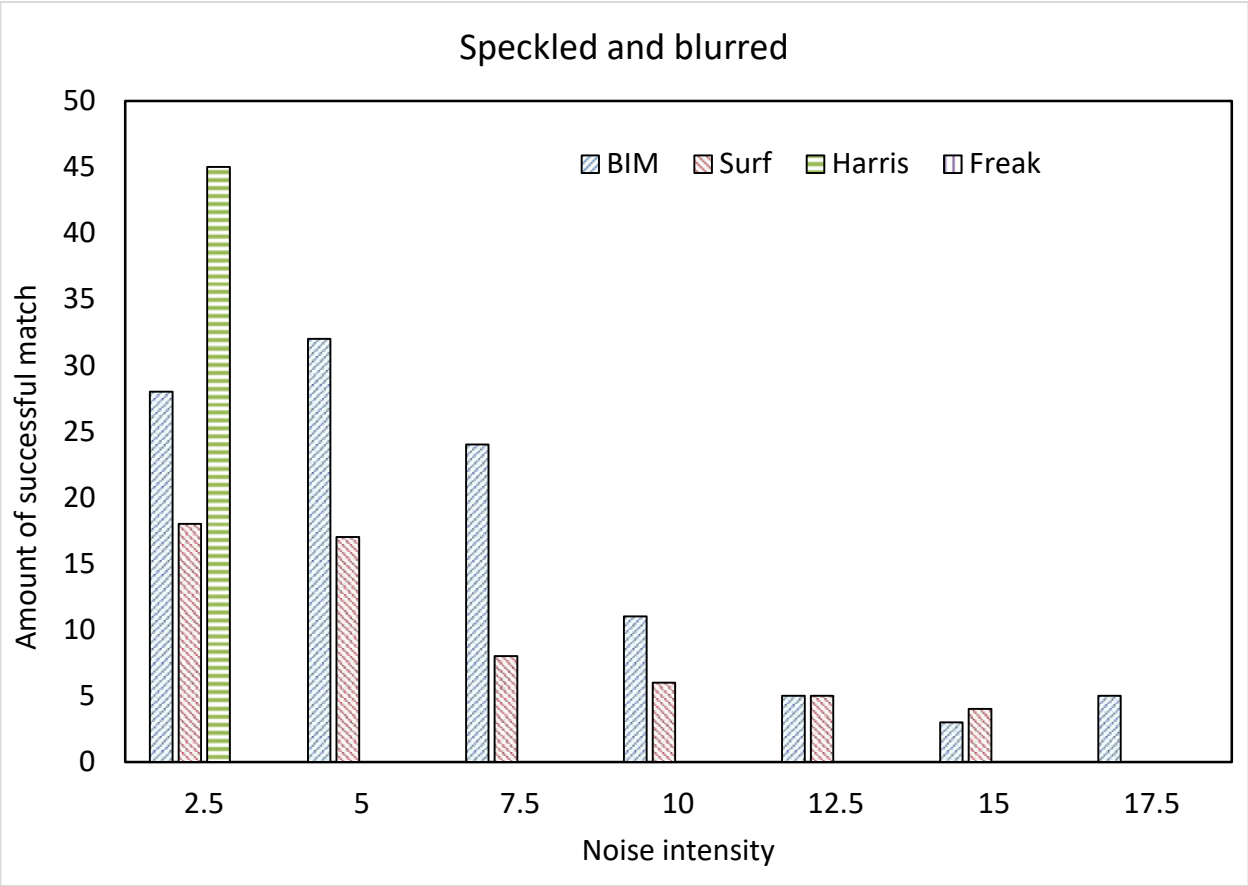


Figure 54 Amount of image successfully stitched depending on salt and pepper intensity, with a sample of image at the maximum noise tolerance.



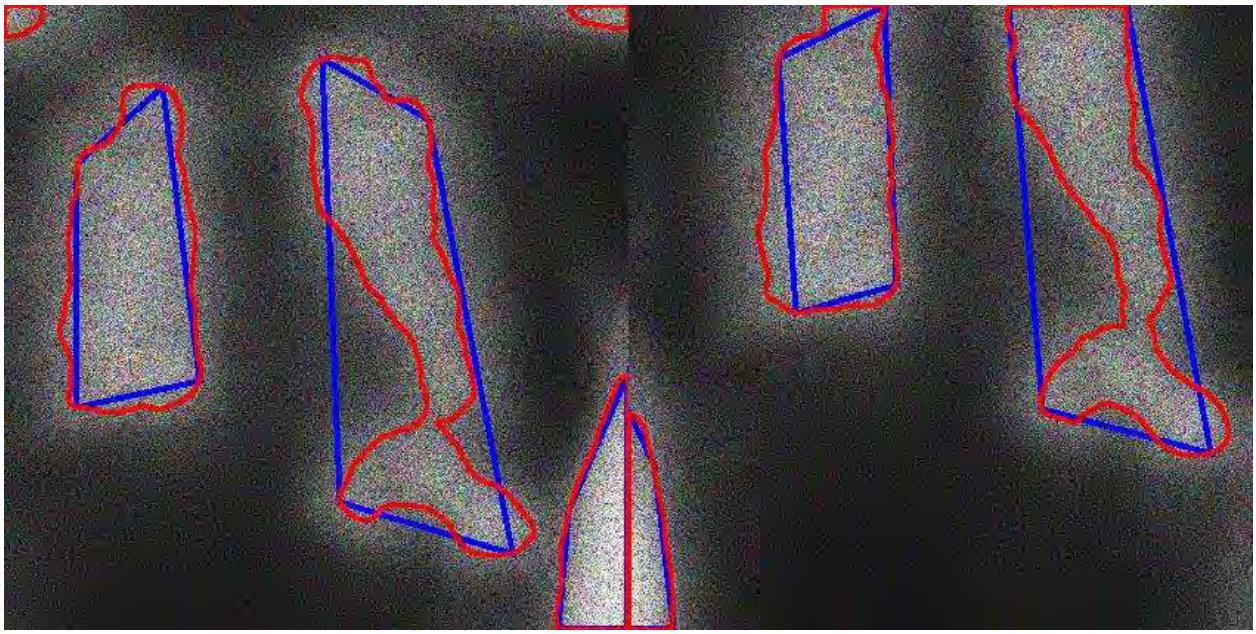


Figure 55 Amount of image successfully stitched depending on speckle and blur intensity, with a sample of image at the maximum noise tolerance.

3.8 Specific use cases of BIM

3.8.1 Feature comparison through different noises

Figure 56 shows an original image (Figure 56.1.a.) and its thresholded equivalent (Figure 56.1.b.) after the steps described in 3.6 BIM process. The following images (Figure 56.2.a. to Figure 56 6.a.) present the same image than Figure 56.1.a. with different noises applied, as described in 3.7.5 Stitching success rate and their thresholded equivalent (Figure 56.2.b. to Figure 56.6.b.):

- Figure 56.2.a., Salt-and-pepper noise
- Figure 56.3.a., white overlay and lenses partial obstruction
- Figure 56.4.a., multiplicative and blurring noise
- Figure 56.5.a., brightness correction (negative)

Figure 56.6.a., brightness correction (positive)

Figure 56 demonstrates that it is possible, using BIM, to compare images presenting different kind of noises. The figures highlighted in Figure 56.1.b. shows five features detected using BIM, All the following images (Figure 56.2.b. to Figure 56.6.b.) present one or more of those features, making the comparison possible.

BIM is the only tested technique registering successes in similar comparison. Such comparison can be used to compare images and assess that they have been taken in similar landscape, even if the images have been taken in an interval including different environmental conditions. Such comparison also allows to reconstruct an image out of a series of noised fragment, given a technology comparing images areas quality and selecting the best quality, BIM can overlay the series of images thus ensuring the positioning of fragments in corresponding coordinates.

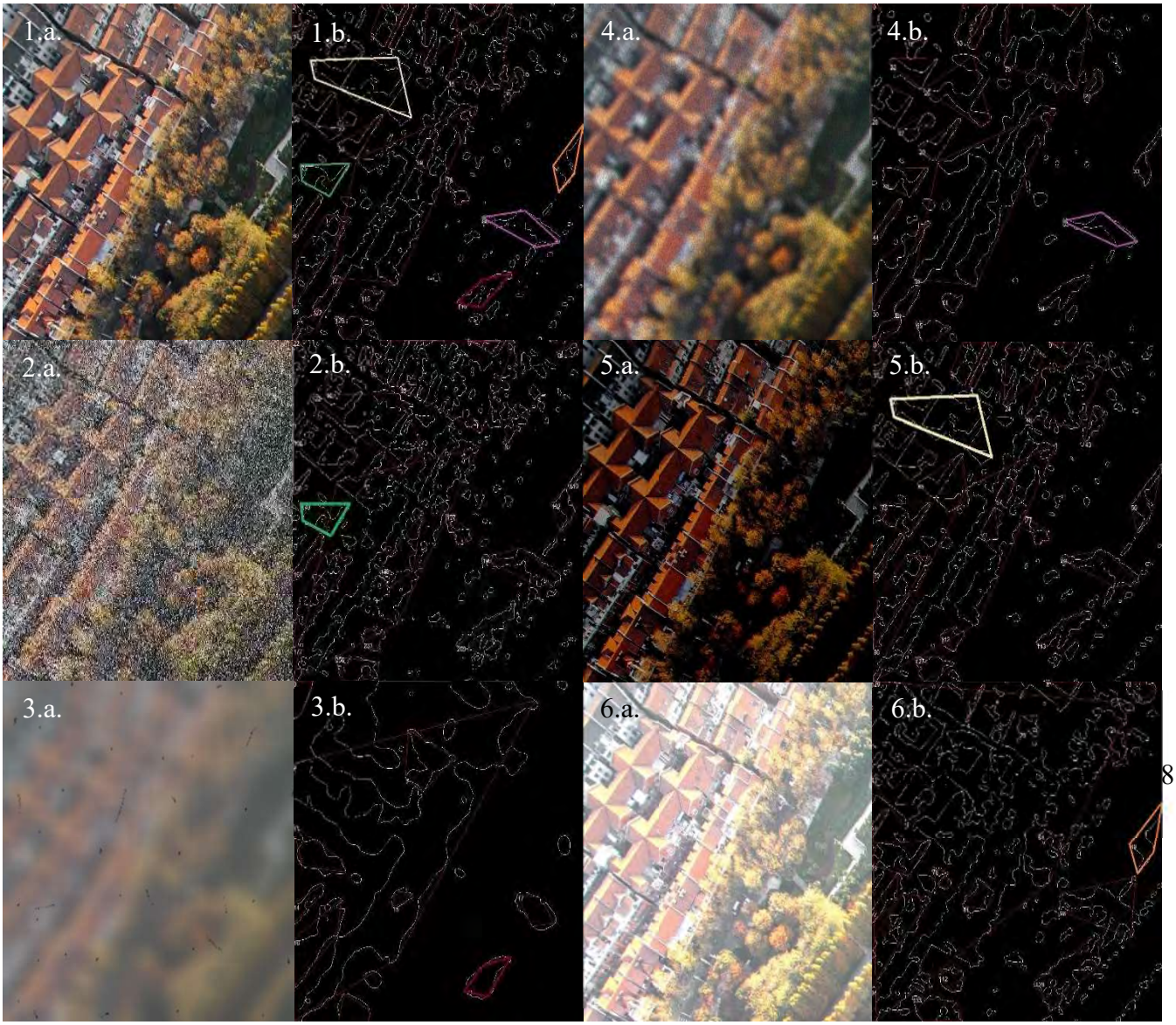


Figure 56 Illustrate an original image (1.a) and the same image submitted to different noise (serie a). Along with their thresholded "print" (serie b). On all images on the serie b, at least one shape (highlighted) can be compared with the original image's threshold.

3.8.2 Stitching large amount of unordered images

Stitching large amount of unordered images is a common problem (Brown M., 2005) (Au, 2013) the problem occurs when a dataset is given and images it contains do not have a labelling containing an information hinting the image succession order. Such dataset can also contain unrelated images complicating the comparison. As stated earlier in 3.7.4 Processing time, processing time of unordered dataset grows exponentially for each new image, as each image's features has to be compared to the entirety of the dataset image's features.

Figure 57 shows an application of BIM to reconstruct an aerial view out of 338 images. In this example, 222 images out of the set were selected for the homography, according to the technique described hereafter.

Figure 57 Aerial view reconstructed using 222 images from a set of 338.

BIM allowing to recognize features with an above average level of confidence (more than half of key points detected are on average used as described in 3.7.5 Stitching success rate), the issue is addressed by assuming that a higher amount of key points matched on a part of directly correlates with the chances of creating a successful homography between images. The key points pairs between images are presented as graphs, where the images are vertices, and the numbers of key points edges' weight. Matching key points are detected using Prim's algorithm (Thomas H. Cormen, 2009), a minimum spanning tree (Thomas H. Cormen, 2009)

is built in order to maximize the amount of matching key points between images and select an appropriate cascade order. If for some set of key points pairs the correct projective transformation cannot be established, then the corresponding image is excluded from consideration and the graph is reconstructed accordingly. Stitching is finally carried out in the tree's traversal postorder (Thomas H. Cormen, 2009).

3.8.3 High confidence stitching

As stated above in 3.7.5 Stitching success rate, a higher number of point augment the confidence of creating a relevant homography for the image and reduce the subsequent uncertainty factor in RANSAC (Rahul Raguram, 2009). However, as stated in 3.7.4 Processing time, the amount of points found is a central characteristic of the technique. As such it was necessary to find a way to be able augment, if necessary, the amount of key point to be found on an image in order to augment the system's confidence. Image are treated with different Gaussian blurring and thresholding values instead of the recommended ones only, creating different unrelated blobs instead of one set, then those different blobs are compared to each other.

RANSAC confidence in finding an appropriate combination can be calculated by the following formula (Johan Nysjö, 2013):

Equation 22 RANSAC confidence

$$N = \frac{\log(1-p)}{\log(1-(1-e)^2)}.$$

For e the probability that the point is an outliner, s the number of points in a sample, N the amount of sample and p the probability of selecting a valid sample. Giving the probability p by:

Equation 23 RANSAC outliner calculation

$$1-(1-(1-e)^s)^N = p.$$

The amount of points found grows in pair with the number of pre-processing parameters tested simultaneously as shown on Figure 58, up to a certain point. Depending on the image, near 820 different iteration using different thresholding values, the image produced stops being informative (containing features BIM can match). The time taken by the process depend on the amount of iteration and present a logarithmic growth.

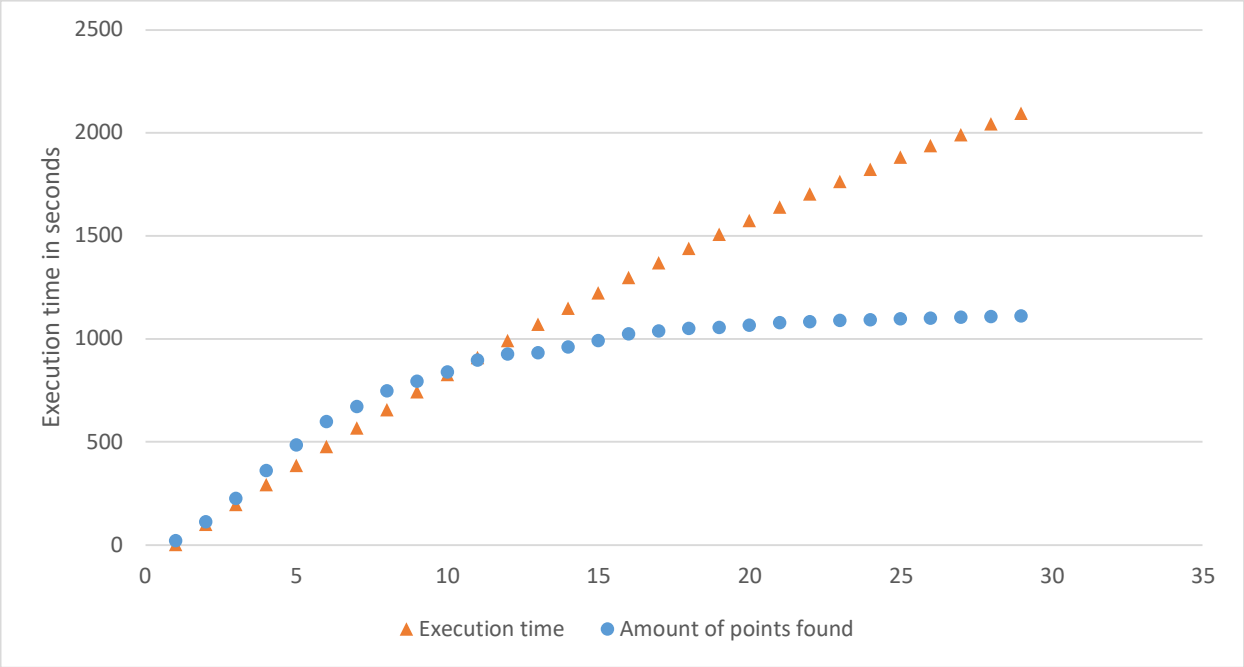


Figure 58 The average amount of points found relatively to the amount of image preprocessing with different characteristics.

To select points, all combinations are first calculated using different pre-processing parameters, in an amount depending on the confidence necessary. Combination are picked from the recommended values as described in 3.6.5 Gaussian blurring and 3.6.6 Thresholding and then in its surrounding with an increasing α variance. Resulting in an array as presented on Figure 59, where all possibilities between parameters are showed with on the X axis, Gaussian blurring radius and on the Y axis thresholding value.

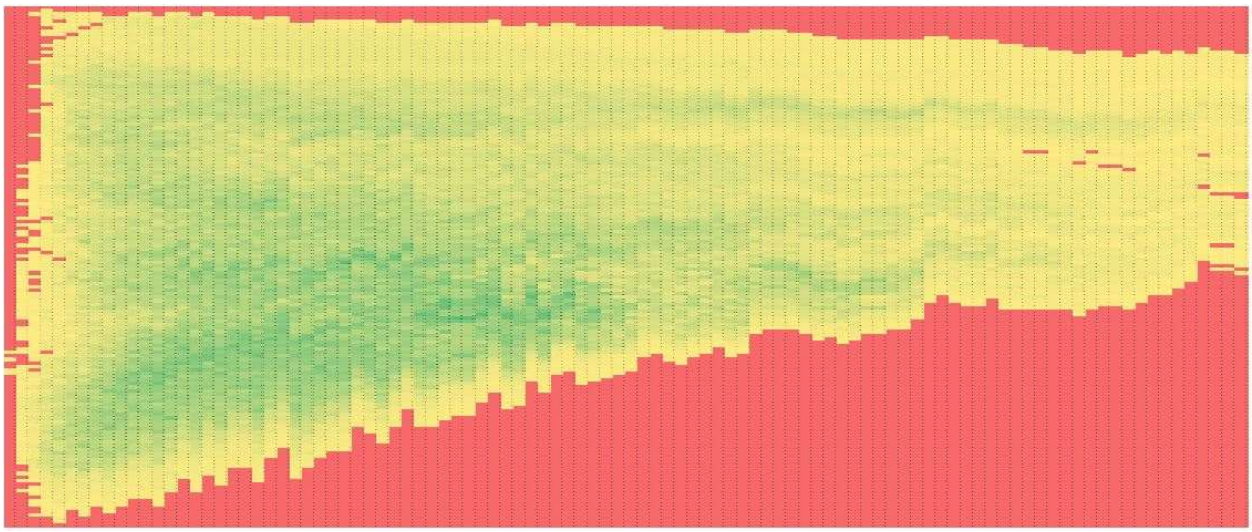


Figure 59 Array of feature selection with different preprocessing parameters. The closer to green, the more combination are found.

Feature are then filtered according to their proximity, if two features from the same image can be compared between each other from a parameter set to another, the feature is

deleted. This avoid cumulating similar points, which would only be a noise source as in increases de possibility of a false positive while creating areas with an accumulation of points extremely difficult for the RANSAC algorithm to dissociate. Once filtered, points are diminished by an average of 99.4% as presented on Figure 60, leaving only the points ideal for comparison.

The last step is running the points found through the RANSAC algorithm presented earlier, resulting in a relevant array of points left for the homography matrix.

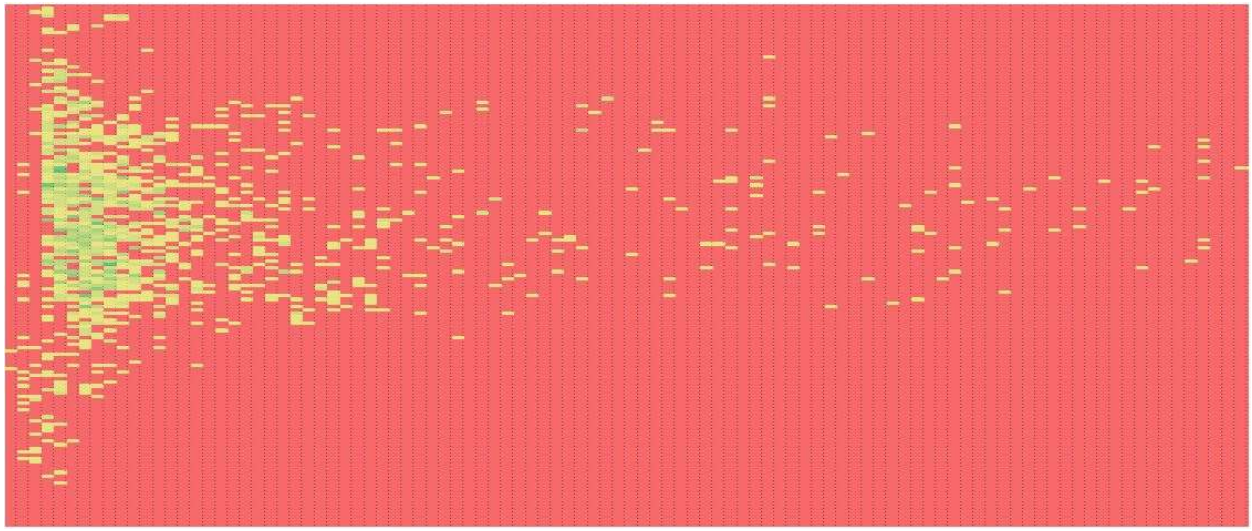


Figure 60 Array of feature selection with different preprocessing parameters after filtering as described above. The closer to green, the more combination are found.

3.9 Conclusions on BIM

3.9.1 Aims achieved

BIM showed higher performances in stitching noised images than any other techniques, scoring almost three time more successes than the second best performing one, among other features. As such, the technique achieves the stated aim.

3.9.2 Tasks solved

BIM was originally developed to improve feature selection and comparison on noised image sets. The benchmark used was to select a technique able to function under different noise conditions and exceed by twice the performance of existing technique to a given dataset as described in 3.7.3 Results evaluation. The of 62% success rate have been matched and exceeded, reaching a success rate of 65%. Moreover, BIM when above expectations on non-noised dataset performing, with a success rate of 91.8%, only 0.4% worse than the most successful technique, making BIM second out of 4. Moreover, the technique's processing time remains inferior to other techniques thanks to its feature selection process presenting a radically different approach from existing techniques. The reduced amount of highly qualitative features is as demonstrated also useful, with a use of it already demonstrated for the stitching of massive number of images. The Such features the allows to position BIM among main existing feature selection techniques while distinguishing the technique by its unique process among them.

3.9.3 Statement

The work presented in this chapter proves the relevance of the technique implemented through a series of comparative tests corresponding to a scientific techniqueology. The technique allows solving feature detection in noised images with higher performances and unique features when compared to existing techniques, proving BIM's relevance when applied to noised images.

3.9.4 Final word on BIM

The reduced amount of feature selected, and the quality of those features is to be taken in account and might find specific uses. In general, the technique is expected to be used for aerial captures as it allows a flying drone to perform stitching, disregarding meteorological conditions and noise inherent to altitude.

The technique uses already known approaches for data processing, making its implementation easy to perform in terms of engineering. In general, the feature selection process application as proposed is a complete success in terms of selection on images (BIM).

4 Key point detection on time series

The aim of this thesis is key point detection on noised data, the first part of this thesis describes the algorithm proposed, following by an implementation proposition on noised images. From this chapter, the same algorithm, which is the innovation proposed in this thesis is applied to noised data series. This chapter introduces the problem, starting by describing existing techniques, followed by the solution developped in the framework of this thesis and a description of the results obtained.

The objective presented in this chapter is similar to the one presented in the previous chapter, with as only significant difference the number of dimensions presented in the problem. The objective here is to find on two time series $S(x)$ and $S'(x')$, points (x) on S and (x') on S' corresponding to the same periodical feature. With a minimal dependence to the prior presence of noise on the series. As such the mapping $(x) \rightarrow (x')$ researched should satisfy consistency constraints and minimize the energy cost represented $E(S, x, S', x')$ as in its simplest form:

Equation 24 time series point matching minimized energy cost

$$E = \|S(x) - S'(x')\|.$$

4.1 CERN and SmartLINAC project

The issue behind the development of SNiF emanated from the European Organization for Nuclear Research (CERN). CERN, based in Switzerland, is a research organization operating world's larger particle physics laboratory. It is home to the Large Hadron Collider (LHC), which is currently the most powerful particle accelerator and largest machine on earth. The laboratory's achievement includes the creation of the World Wide Web (WWW) in 1989, the first production of antimatter in 1995 and the discovery of the Higgs boson in 2012, completing the standard model and receiving the following year the Nobel Prize in physics.

CERN tackles constant challenges in the field of IT, having one of the world's most demanding computing environment, having to handle the one petabyte of data produced by the collisions observed through the LHC's experiments (CERN, 2018). CERN's datacentre currently stores several hundred of petabytes of data and expects the High-Luminosity Large Hadron Collider (HL-LHC) to multiply this amount and enter the Exabyte regime (Di Meglio, 2017). The Worldwide LHC Computing Grid (WLCG) makes the process of data analysis possible, making available 770'000 computer cores on top of CERN's 230'000 for the calculation of the data obtained by particle collisions.

In this context, CERN openlab manages the development of ICT solution for the LHC community and wider scientific research through collaborations with leading companies and research institute. Samara National Research University signed in 2018 a collaboration agreement with CERN openlab, followed in 2019 by the birth of the SmartLINAC project.

Linear Accelerators (LINACs) are a type of particle accelerators able to accelerate charged particles to high speeds. They are used today in many different researches, industrial and medical applications, from particle physics research, to cancer treatments, non-destructive material testing, nuclear waste treatment, security screening, or food sterilization. Typical medical or industrial LINACs are complex engineering systems and their operations, especially for in clinical environments, are highly impacted by down-time, costs of operations and lack of trained engineers. They are complex engineering systems composed of hundreds of thousands of parts, subject to continuous operations and are naturally subject to failures and breakdowns. In situations where systems failures have major safety or economic impact, it is of paramount importance to understand the system failure modes at the components level and design efficient maintenance plans allowing to maximize the up time, decrease operational costs and limit unexpected incidents. Very often the maintenance operations are reactive rather than proactive, which increases the operational costs, increase down-time and may force to keep expensive spare parts and trained engineers available at all times.

the complexity of such systems is today severely limiting the availability and diffusion of LINACs for medical applications, technical expertise is not available, and down-time impacts directly the patients' life expectancy. The need for simpler-to-maintain-and-operate medical LINACs was highly stressed during a workshop jointly organized by CERN, the International Cancer Expert Corps (ICEC) and STFC in October 2017 (David Pistenmaa, 2017).

The SmartLINAC project investigates an innovative technique to understand and model the failure of LINAC systems down to the individual component level, design predictive, easy-to-implement maintenance plans based on Mean-Time-To-Failure and operating conditions' information. Then to dynamically refine and "personalize" the maintenance plans by analysing information extracted from production systems using a machine-learning-based approaches.

The project's is being pursued in its first phase using CERN's LINAC 4 operation data, in particular RF power source outputs, which are representative of the data obtained from other

LINAC components and seemed, according to domain experts, to be directly related to beam degradations in CERN's accelerator complex.

Linear accelerator 4 (LINAC 4) will be from 2021 the source of LHC's proton beams. Designed to boost negative hydrogen ions (a hydrogen atom with an additional electron) to high energies (160 MeV) in order to prepare them for the Proton Synchrotron Booster, starting their journey through the accelerator complex, as represented on Figure 61.

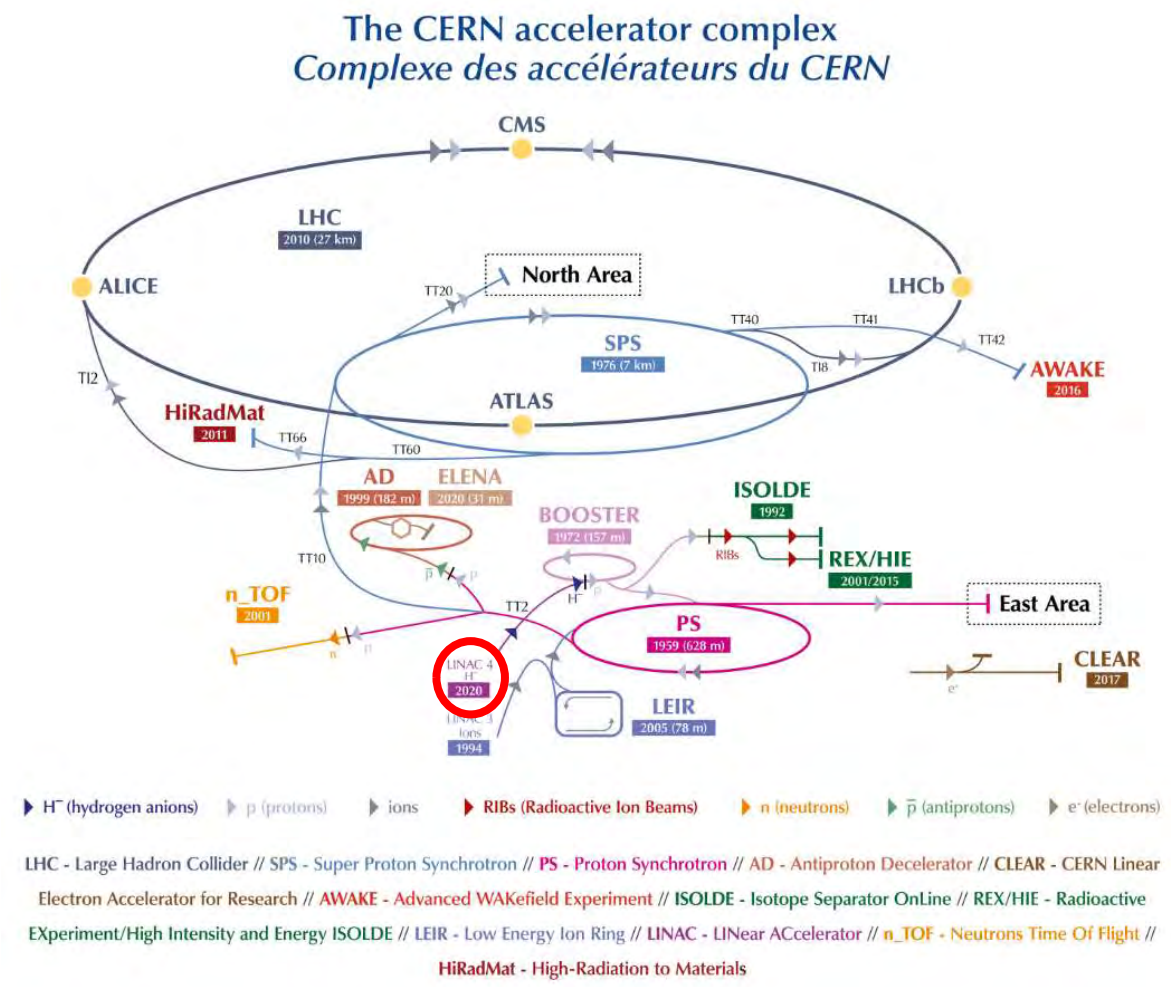


Figure 61 Represent CERN's accelerators complex with, highlighted in red, LINAC 4's position, at the beginning of the injection chain (CERN, 2019).

The Radio Frequency was LINAC4's most common downtime cause during its reliability run, cumulating more than 36% and the reduction of down time periods have been established as a priority since 2018 (O. Rey Orozco, 2018). It is situated at the beginning of LINAC4's functions in low energy a shown on Figure 63, on the third position.

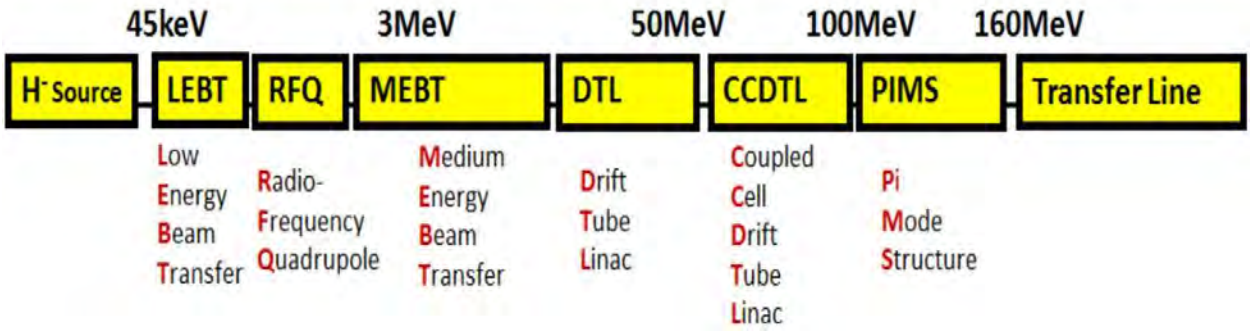


Figure 63 represents LINAC4's basic architecture as represented in "PERFORMANCE EVALUATION OF LINAC4 DURING THE RELIABILITY RUN" (O. Rey Orozco, 2018). The RF source is represented in the graphic's third position. With illustration of the beam acceleration for each phase in Mega electron-volt (MeV).

During the 13 weeks of reliability run, LINAC4 showed 90.6% of uptime, counting a total of 387 faults with a mean time to repair of 43 minutes. Main LINAC4 downtime root causes are shown on Figure 62, with power converters being second, partially due to a single 18 hours downtime over two faults.

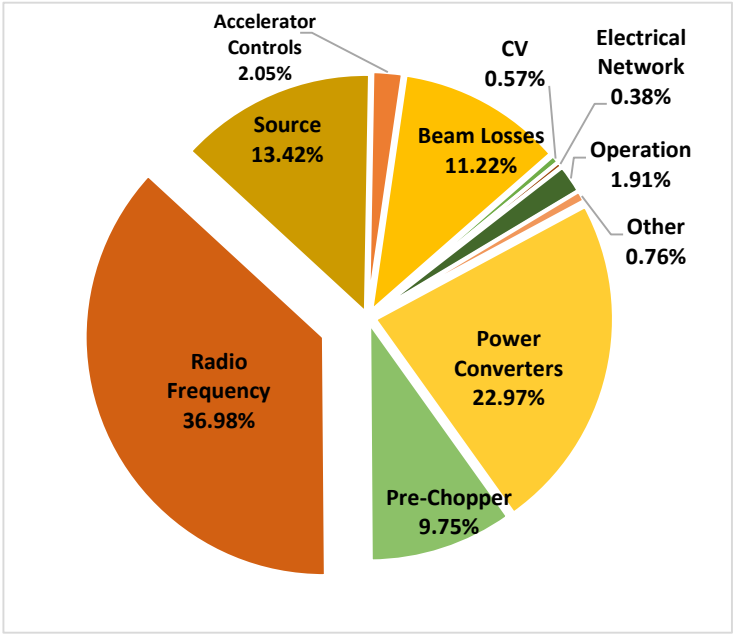


Figure 62 LINAC4 root causes system faults time proportions during the first two phases of reliability run (O. Rey Orozco, 2018).

4.1.1 Explanation of the data

LINAC 4 uses 2MHz RF sources to produce de plasma from which particles are extracted to the beam. The data consists of the source's output power in Watts over time as represented on Figure 64 The figure represents data during a run of several months, parts highlighted in blue were marked by a domain expert as period when the beam quality presented decay. It coincides with periods of jitters in the data source. It is not excluded that those periods are the only periods of beam quality decay, other than that information, no form of labelling is obtainable from the data. In such conditions, the project's priority was to establish a relevant data labelling technique relative to the periods of anomaly in the beam, which would further allow the selection of area of interest for the eventual anomaly prediction.

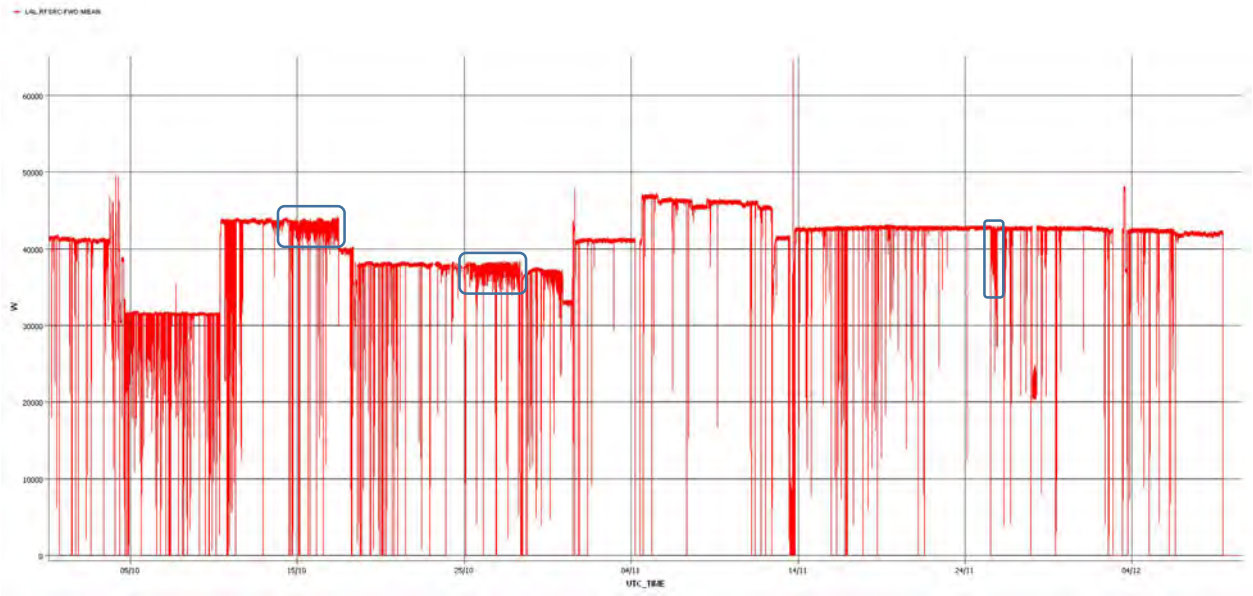


Figure 64 LINAC 4's RF power output from the autumn run 2018. Portions highlighted in blue by a domain expert as periods where the beams quality presented decays (Yann Donon A., 2019).

4.1.2 Description of noises

The data itself presents several sources of data driven noises, that had first to be understood and should be treated differently depending on their source. It was important to evaluate the informativeness of each noise before classification. This chapter details each sources of noise and their processing. The data analysed never presented breakdowns of the components but only changes of behaviour (beam quality decay), the investigation focuses therefore on any data distant from a regular behaviour that correlates with indicated periods of beam quality decay and close anterior period that could indicate symptoms.

In the data analysed, some measurement register 0W, leading to the drops visible on the bottom part of Figure 64 as well than highlighted in Figure 65.1.a. Those drops are consistent through the whole data set, with some rarer occurrence being registered at half of the power as in Figure 65.1.b. Following those characteristics, they can be considered as intermittent black noise as it registers systematically a null power no matter the signal state, however the signal is not mainly black (null), making it intermittent (National Telecommunications and Information Administration, 1949). Such noise comes from

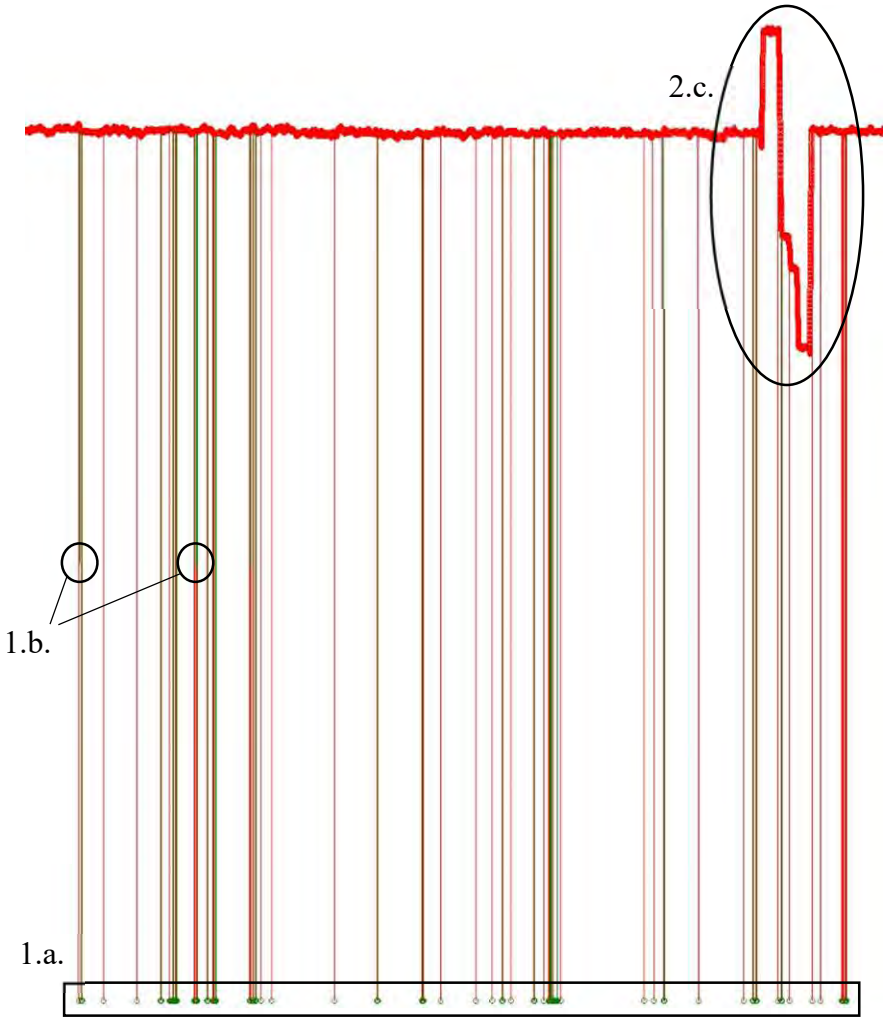


Figure 65 Sample of data over a 1-day period with in 1.a. 0W captions, 1.b. punctual registration of power drops, 1.c. manual punctual power modification (Yann Donon A., 2019).

measurements errors and drops in the source.

The reason of behaviours observed in Figure 65.1.a and Figure 65.1.b remains unexplained, their distribution is relatively even and doesn't present any form of correlation with the designated periods of jitters. Those features led to a principle of punctual outliners elimination. As such, all the points presented in green on Figure 65 were considered of this

category. Such point was dumped from the data series, connecting their previous and following points.

Another source of observable noise in the analysis is presented on Figure 65.2.c and detailed on Figure 66, such noises results from human interactions with the data source, they can occur for different reasons, Figure 66.1. shows consecutive selection of decreasing power values for evaluation purposes on the beam. Figure 66.2 shows a student peak corresponding to a stress test and Figure 66.3 a regular change in power. Such event could not be related to the appearance of decay in beams quality however it has been on several occasions used to interrupt the jittering periods, resulting in jittering periods often ending with power source modifications. This can be assimilated to burst noise (Texas Instruments, 2007) sometimes combined with oscillator phase noise (Thomas H. Lee, 2000).

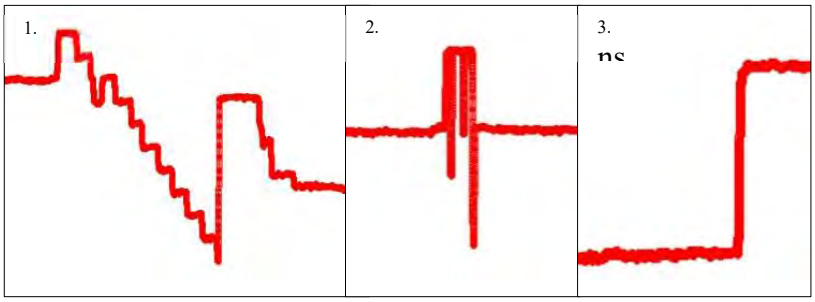


Figure 66 Examples of power modifications resulting from human operations on the source (Yann Donon A., 2019).

Another source of noise is the general volatility of power in observed samples, as illustrated on Figure 67. This noise is directly issued by the source and the measurement instruments used. It is visible in the image that samples varies largely from a point to the next, making difficule to extrapolate informations from the data without preprocessing. This high variance, or white noise (Diebold, 2007), is a prime source of false positive and negative when data are computed using existing techniques. Moreover, the average variance doesn't show any consistency whether in period or normal operation or jittering periods when selected on sufficiently small windows. Experiments on a 100 windows of 90 samples showed an average variace varying from a extreme to the other from simple to more than 7, effectively ranging from an average window variance of 42W to 251W. Moreover, as white noise, it is independent from normal operations to periods of jittering with a sample distribution even in all frequencies observed.

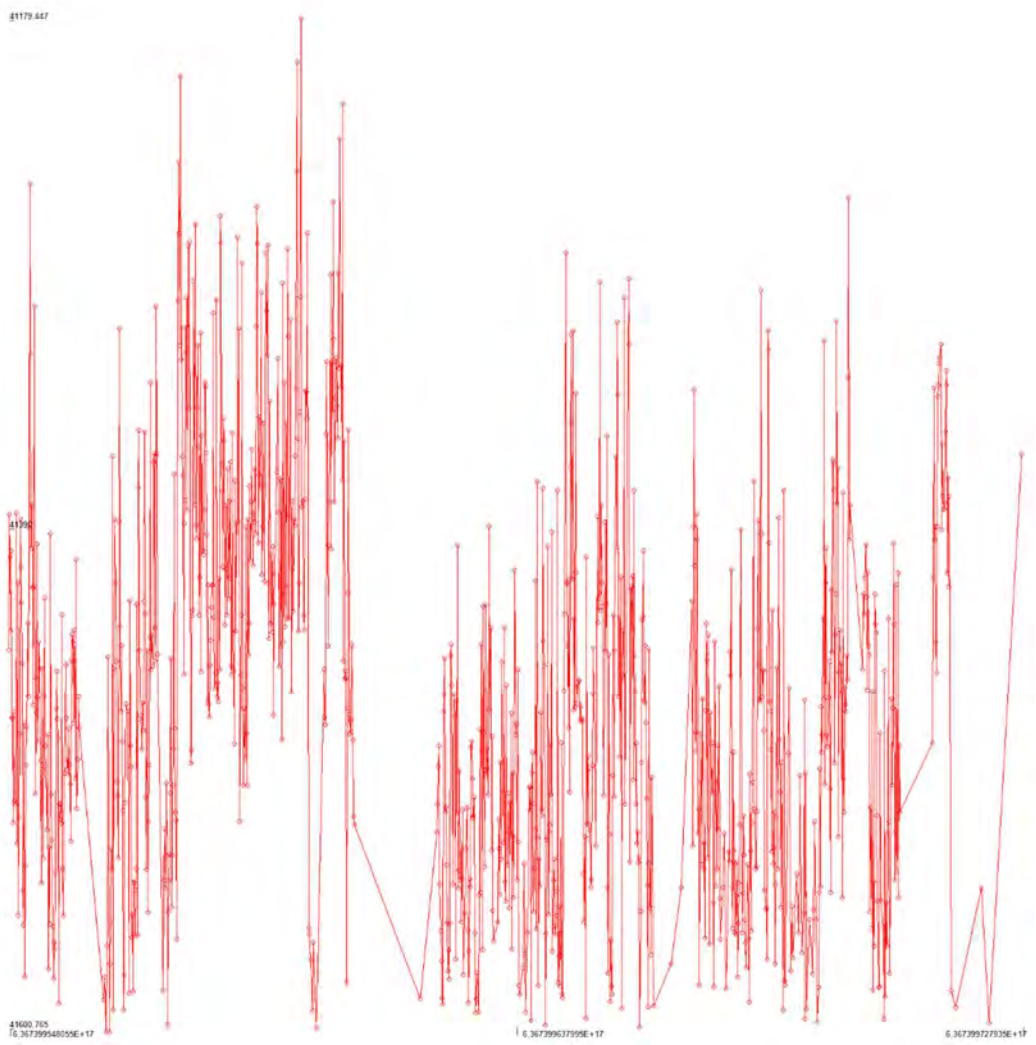


Figure 67 shows a data samples over a few hundred observations. It highlights the general variance observable in the data.

Description of the jitters

Jitters are the features that should be recognized and isulated as they correlate with periods of beam decay in the accelerators.

Affectivey, jitters are consituted of successions of violents peaks and drops. Those peaks and drops however can not be considered as a reference alone as they freaquently occurres in the signal. It is their accumulation that characterises a period of jittering.

The jittering periods can be easily observed on the signal over time. The following Figure 69 shows a one-week period of normal operation and the same time length of jittering.

4.2 Existing techniques

As for BIM, existing techniques have been tested on the data presented above. The issue systematically encountered wasn't to detect the jittering period themselves. As introduced in Figure 2, all techniques succeeded to some extends, the issue encountered is the amount of false positive and false negative. Even in relatively small quantities, it represents a relevant issue for labelling and differentiation of the signal state. Moreover, existing techniques are all heavily impacted by noise. False positive are due to all source of noise described in 4.1.2 Description of noises, as observed in Figure 2.

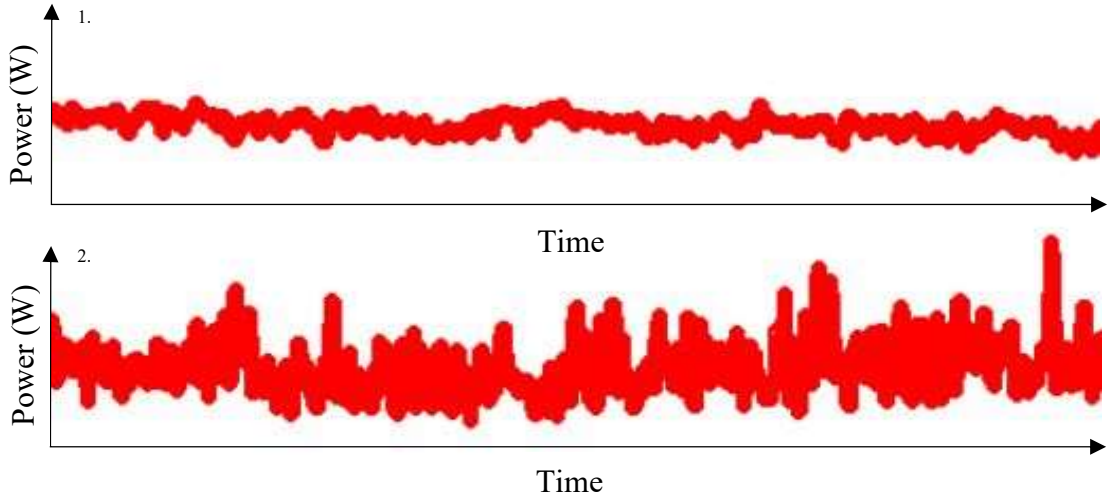


Figure 69 Power signal over normal operations (1.) and jittering period (2.) over a week of time (Yann Donon A., 2019).

All techniques are compared using the following references, categorisation are selected according to approximations established in 4.1.1 Explanation of the data :

- True positive, point labelled as jittering and being effectively jittering
- True negative, point labelled as a normal period of operation and being effectively so
- False positive, point labelled as jittering when it doesn't correspond to a jittering period
- False negative, point labelled as a normal period of operation and corresponding to jittering periods

The jittering being effectively continuous during jittering periods, labelling should be accordingly consistent for series of points

4.2.1 Label-related clustering

This approach is characterised by the use of machine learning in an attempt to solve the problem. Labelling was established based on the estimation offered by domain expert as developped in 4.1.1 Explanation of the data. Figure 71 shows a visualisation of the data with in red four areas used as jittering period reference.

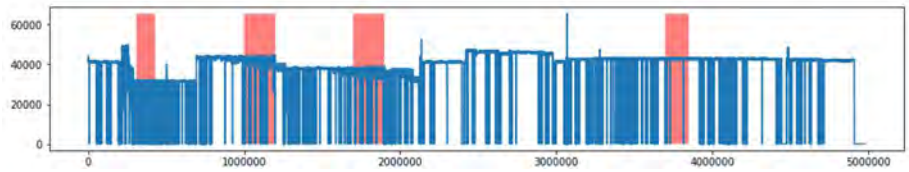


Figure 70 Training data: RF power sources output, red color shows four jitter areas (Yann Donon A., 2019).

Figure 71 Shows jittering areas in red as displayed in Figure 70 Training data: RF power sources output, red color shows four jitter areas . with in blue proximity areas showing feature sets comparable to the ones observed in jittering areas (Yann Donon A).

The technique is based on the search of features distinguishing the marked interval from the rest of the data. Features are observed in windows and "proximity areas" showing comparable features to the jittering areas windows are selected as displayed on , where proximity zones are highlighted in blue around the jittering periods

This approach leaved, surrounding the jittering areas large areas potentially assimilated to jitters, creating false positive observations. Moreover, fragments are non-monolithic as shown on Figure 2, meaning they are prompt to false negative labelling.

Label-related clustering technique sets if windows are clustered using Kernel Density Estimation (KDE) (Rosenblatt, 1956; Parzen, 1962). The Adjusted Rand Index (ARI) (Arabie, 1985) is then calculated between the clustered set and the labelled set. ARI values are categorized according to a threshold t with corresponding subsequences. This technique showed itself relevant ant the use of KDE and ARI for scalability showed the feasibility of realizing an adaptable system. Figure 72 illustrate the labelling offered by label-related clustering technique, compared to the signal (black) and the labelling estimation (red).

Results have been obtained with a threshold value set at 4.1, which showed the optimal performance in our experiments, resulting in the estimation displayed in Table 1.

	True positive	True negative	False positive	False negative	Total
Absolute amount	1046	787852	1756	217708	1008362
Proportion (%)	0.1	78.18	0.2	21.6	100

Table 1 represents the label-related clustering technique results on a set of 1008362 entries.

It is observable from Figure 64 and Table 1 that the technique display a significant

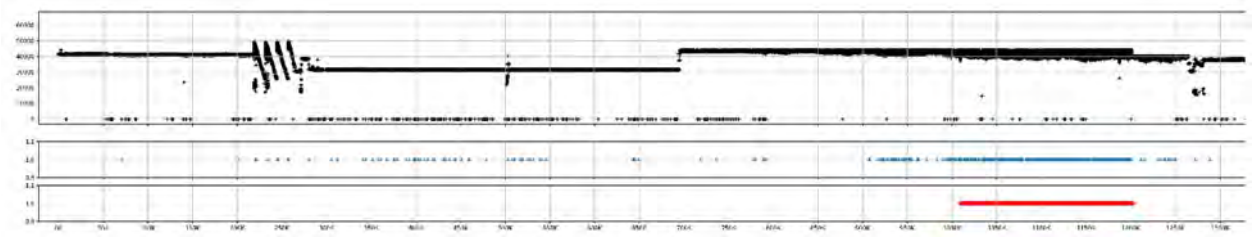


Figure 72 Illustration of the labelling obtained from a sample of LINAC 4's data with in black, the signal, in blue the labelling obtain from label-related clustering and in in red labelling estimation by domain expert

sensitivity to noise as jittering symptoms are detected over the whole sample in non-labelled locations. However, it is relevant to underline that a strong concentration of data labelled as jittering symptoms appears in the "proximity area" of the anomaly, highlighting symptoms of anomalies. The technique altogether lacks of the consistency through jittering periods as developped in 4.2 Existing techniques but shows relevant information over the data themselves.

4.2.2 Sequence analysis using statistical features

As introduced in the article "Extended anomaly detection and breakdown prediction in LINAC 4's RF power source output" * (Yann Donon A., 2019): This technique consists in processing the sequence by sliding window and calculating the statistical features for the fragments of the sequence located in this window (Mayur Datar, 2012).

The idea behind the approach is based on the assumption that there are some statistical characteristics allowing predicting the appearance of abnormal periods in time series (anomalies).

As it is shown in 3.1, the transition between the normal and abnormal state does not occur instantly, meaning the sequence does not only contain normal and abnormal intervals, but also transition stages. Meaning the detection of such transition stages can be used predict anomalies. The exact amount of transition intervals being unknown, clustering algorithms must be used to determine their number and characteristics.

Thus, the problem is reduced to the division of the initial sequence into N clusters based on the values of statistical features. Processing of the sequence will be carried out using a sliding window of size L with a shift K. the Features are the statistical features of the sequences: mean, variance, asymmetry, kurtosis and percentile (B. S. Everitt, 1998). Hyperparameters of this approach are the size of the sliding window L, the value of the shift K, as well as the values of percentiles.

In our test samples, the data consisted mainly in stable values. However, the interval between 220000 and 290000 is a period of jitters, which should be predicted. Standard deviation can be used to determine the areas containing deviation from the mean value, but because the value from which the deviation is to be estimated varies (intervals with stable values are characterized by different intensity values), the standard deviation must be normalized to the mean value (coefficient of variation (Salkind, 2010), the ratio of standard deviation to the mean value). Assessment of the value of the coefficient was conducted on a sample by a sliding window using the value of the radius of the window is 10 (L = 20), with a shift K=1 (for best accuracy). The maximum value is 4.47. The resulting distribution of the coefficient is shown in Figure 73, with an interval of 0.01.

The main peak at point 0 is the number of values corresponding to the normal intervals

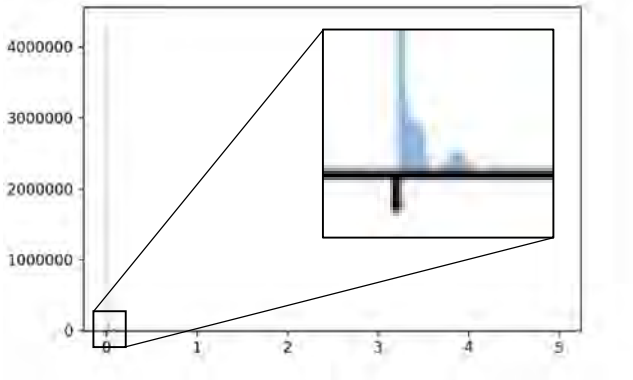


Figure 73 Coefficient of variation distribution graph.

without jitters. Figure 74 shows a more detailed distribution graph in the range [0.01 – 0.2] with two distinguishable intervals in the range [0 – 0.05] and [0.06 – 0.14]

After analysing the graph, it can be can assumed that the fragment of the graph in the range [0 – 0.05] corresponds to normal intervals – "cluster 0". Values (0.05 – 0.15) – cluster 1 – possibly close to the normal intervals, but slightly different behaviour in its instability. Further examinations allow to distinguish 3 more clusters in range [0.2-0.5] and the [0.5-4.5]

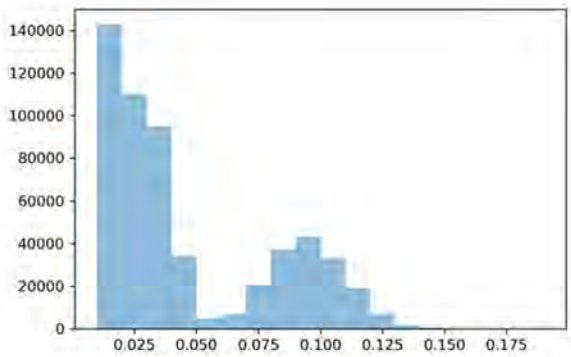


Figure 74 Fragment of coefficient of variation distribution graph [0.01-0.2] containing two clusters (Yann Donon A.)

interval 16 more.

All analysed clusters can show symptoms of jittering periods, the classification technique allowed distinguishing 20 distinct clusters confirming the presence of relevant features observable in the data and delimiting the different sequences, this technique is therefore important for the analysis of anomaly symptoms. The technique shows good performances in detecting anomalies period as highlighted in Figure 75. However, the major issue encountered using this technique is the clusters classification, leading in all sources of change in the signal containing potential symptoms of jittering leading to areas described as noise in 4.1.2 not being distinguished from anomalies, as illustrated in Figure 75.

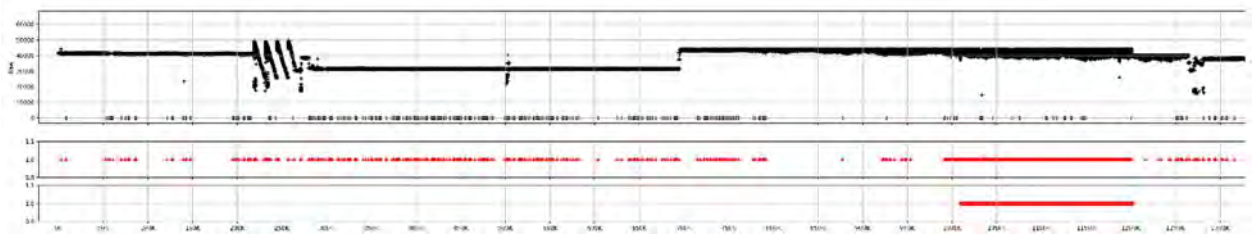


Figure 75 Illustration of the labelling obtained from a sample of LINAC 4's data with in black, the signal, in blue the labelling obtain from Sequence analysis using statistical features and in in red labelling estimation by domain expert.

Characteristics mentioned above are confirmed by the numbers presented in Table 2. The sequence analysis using statistical features presents a higher false positive degree than label-related clustering approach but scores better in the other categories.

	True positive	True negative	False positive	False negative	Total

Absolute amount	8321	775891	6807	208609	999628
Proportion of the total (%)	0.8	77.6	0.7	20.9	100

Table 2 represents the sequence analysis using statistical features technique results on a set of 999628 entries.

4.2.3 Kalman filtering

Kalman filter performs well describing highly volatile series (Brown R. G., 1992), it takes in account features rarely used in other techniques (Shumway R.H., 2017), such as the variance of the initial state estimation and the model error variance (S., 2011). It provides information about the quality of the estimation by estimating an error probability. Kalman filter applies well to real-time digital processing (Lim, 2016) because of its recursive structure allowing execution without storing observations or past estimations (C., 2011), the technique uses smoothing as part as its functioning as shown on Figure 76, which represents in blue an original data sample and in red the corresponding sample after smoothing.

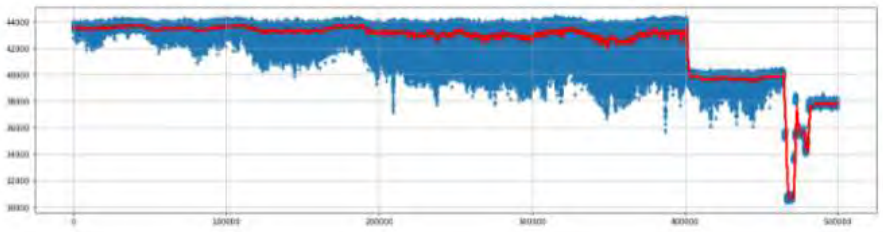


Figure 76 Data fragment before (blue) and after (red) Kalman filter (smoothing) application (Yann Donon A.).

The technique could distinguish two clusters of data according to the data average deviation, as shows Figure 77. Figure 77.1. contains a data sample from the signal after Kalman

Figure 77 Shows a data sample (1.) followed by the first cluster containing supposed jittering periods highlighted in red (2.) and the second cluster supposed to contain anomalies provoked by an action from the users highlighted in red in 3.

filtering. Figure 77.2. highlighting the first cluster in red, representing jitters symptomatic of signal quality decay. Figure 77.3. on the other hand highlights in red the second cluster,

corresponding to noise established by human manipulations as described in 4.1.2 Description of noises.

The overall technique success is good, as the technique could highlight anomalies with precision as shown on Figure 78 while differentiating the different sources of anomalies. This hinted already, at the time the approach was tested, the possibility of using blurring technique for data analysis as used in BIM's process, this observation leading to the use of a similar process in both BIM and, as described further in this chapter, SNiF.

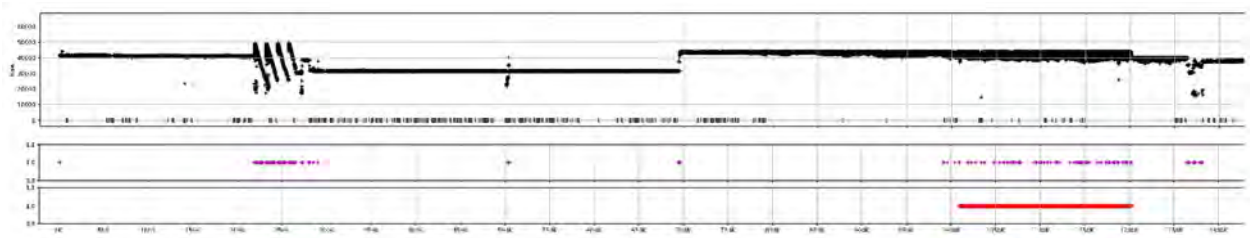


Figure 78 Illustration of the labelling obtained from a sample of LINAC 4's data with in black, the signal, in blue the labelling obtain from Kalman filtering and in in red labelling estimation by domain expert.

Kalman filtering showed good performance and precision taking in account the differentiation of different labels. The technique isn't sensitive to early symptoms as performed by Label-related clustering, moreover, like other techniques, presented above, the absence of labelling continuity is an issue. Table 3 illustrates the metrics proposed by Kalman filtering without taking in account the clustering ability of the technique, thus generating false positive, not taken in account, the technique registered 1980 entries, far under the two techniques presented above.

	True positive	True negative	False positive	False negative	Total
Absolute amount	26273	774832	14776	192159	1008040
Proportion of the total (%)	2.6	76.8	1.5	19.1	100

Table 3 represents the sequence analysis using statistical features technique results on a set of 1008040 entries.

4.3 Series with Noise Featuring

Series with Noise Featuring (SNiF) * (Yann Donon A. K., 2019), * (Yann Donon A. K., 2020) was developped after the initial success showed by BIM. Both technique's initial challenge came from highly noised data which treatment was extremely challenging for existing techniques. The idea occurred after several unsuccessful tests that the same process applied to images could be adapted to times series, first test was immediately encouraging and led to the development of this alternative.

As for BIM presented earlier, this technique was initially thought to process noised series. Already existing techniques showed themselves performant but do not match all the needs related to the problem stated in 4.1 CERN and SmartLINAC project.

This chapter describes SNiF's functioning and makes parallels with BIM steps. Figure 79 echoes with Figure 14, which described BIM process, further comparisons, highlighting the unicity of both processes, will be made in the final chapter of the thesis.

4.3.1 Process presentation

Unlike shown previously, Figure 79 do not present similar illustrations through its process as the nature of the time series used for this research do not allow a visually representative description through one period of time. SNiF's process shows the same numbering approach than the algorithm presented in Figure 3 and BIM's implementation in Figure 14.

SNiF's input is real time data from captors, in our case, the LINAC 4's RF power source. The output offered by the technique should be a labelling and real time alerts when entering a phase of jittering.

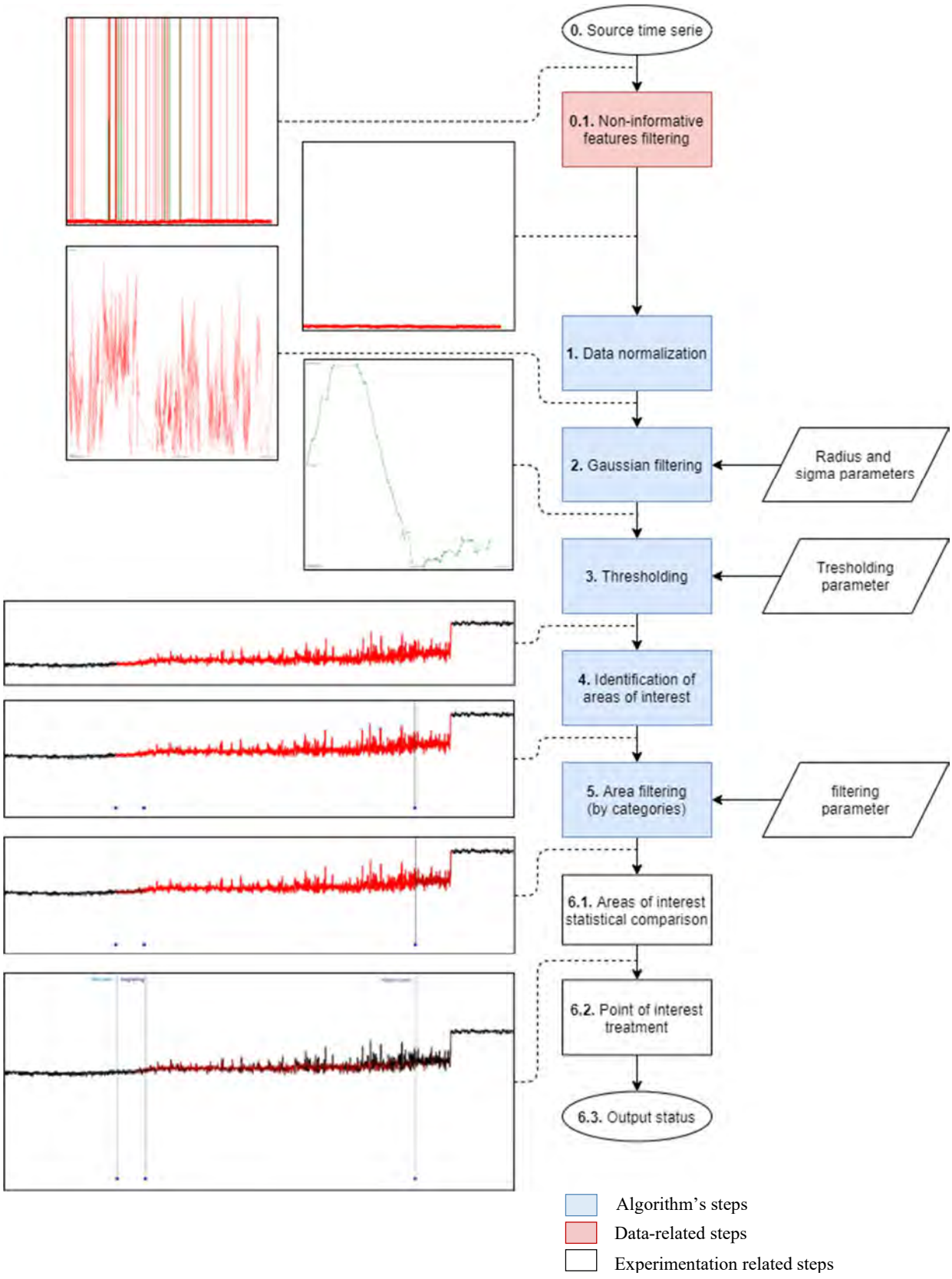


Figure 79 Illustrated SNiF process used for time series analysis

4.3.2 Non-informative features filtering

As showed in 4.1.2 Description of noises some information present in the data are characterised as non-informative. It is relative to the data source itself and a specific kind of noise that is not informative and doesn't influence the signal as developped above therefore, it doesn't correspond to a step of the algorithm presented in Figure 3, it corresponds to Figure 79's step 0.1. step 4 and is identified on Figure 14 as step 4.2.Non informative data are in our research categorised as such when is shown as outlier according to Grubbs' test (Grubbs, Sample criteria for testing outlying observations, 1950). The nature of time series, when analysed in real time made Grubb's test analysis extremely appropriate as the technique assumes 0 to 1 outliner by estimation according to :

Equation 25 Grubbs' test

$$G = \frac{\max_{i=1,N} |Y_i - \bar{Y}|}{s}.$$

For all values of the series Y and $\bar{Y}$ the series' sample mean, and s the standart deviation (Grubbs, Procedures for Detecting Outlying Observations in Samples, 1969).

Filtering outliers allowed to perform further statistical analysis, starting with data normalization. Figure 80 illustrates the same data sample before and after filtering. This step in SNiF process is the only one not having a comparable BIM equivalent as this filtering need is dictated by the data source and not by the data type.

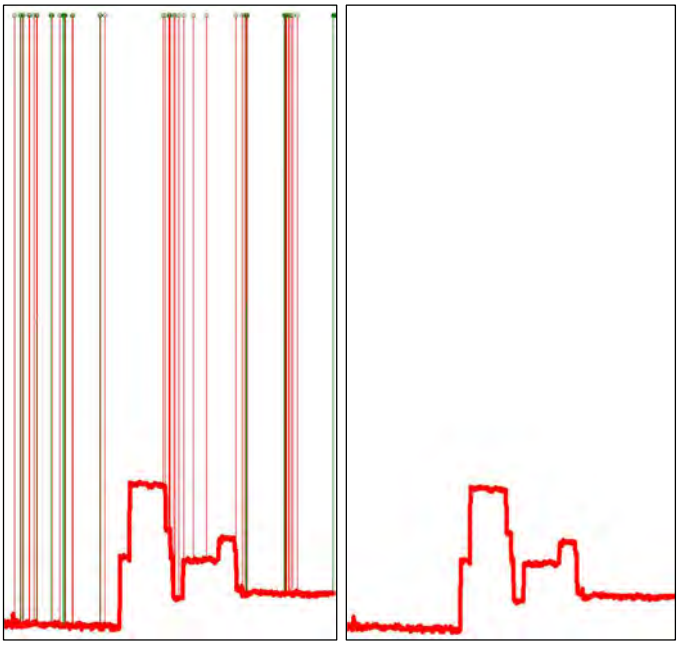


Figure 80 Same fragments before filtering using Grubb's test (left) and after (right) The fragment contains about 50'000 entries, less than 1 ‰ of data were filtered out (Yann Donon A.).

4.3.3 Data Normalization

Data normalization consists of bringing different data on the same scale. Just like histogram normalization described in 3.6.2, this step is used to bring different data samples on the same scale, independently from their value, moreover, it rationalises the data samples after the Non-informative features filtering described in 4.3.2. It corresponds to the algorithm's (Figure 3a) and the technique's process (Figure 79) phase 1.

The technique used is z-score, which preserves the data range and introduce dispersion in the data, which is useful after the filtering realized above. Standard deviation score x is calculated as for μ the sample mean, and σ the sample's standard deviation, z the distance between a sample value and the population mean (Kreyszig, 1979)

Equation 26 Z-score

$$z = \frac{x - \mu}{\sigma}.$$

4.3.4 Gaussian blurring

Gaussian blurring is used to reduce the risk of false positive anomalies detection in the data by changing the signal, initially noised to tendencies. It is represented in the algorithm's (Figure 3) and the technique's process Figure 79.a) as phase 2. It is easily possible to asses visually, as represented on Figure 81, the figure illustrates the same fragment before (left) and

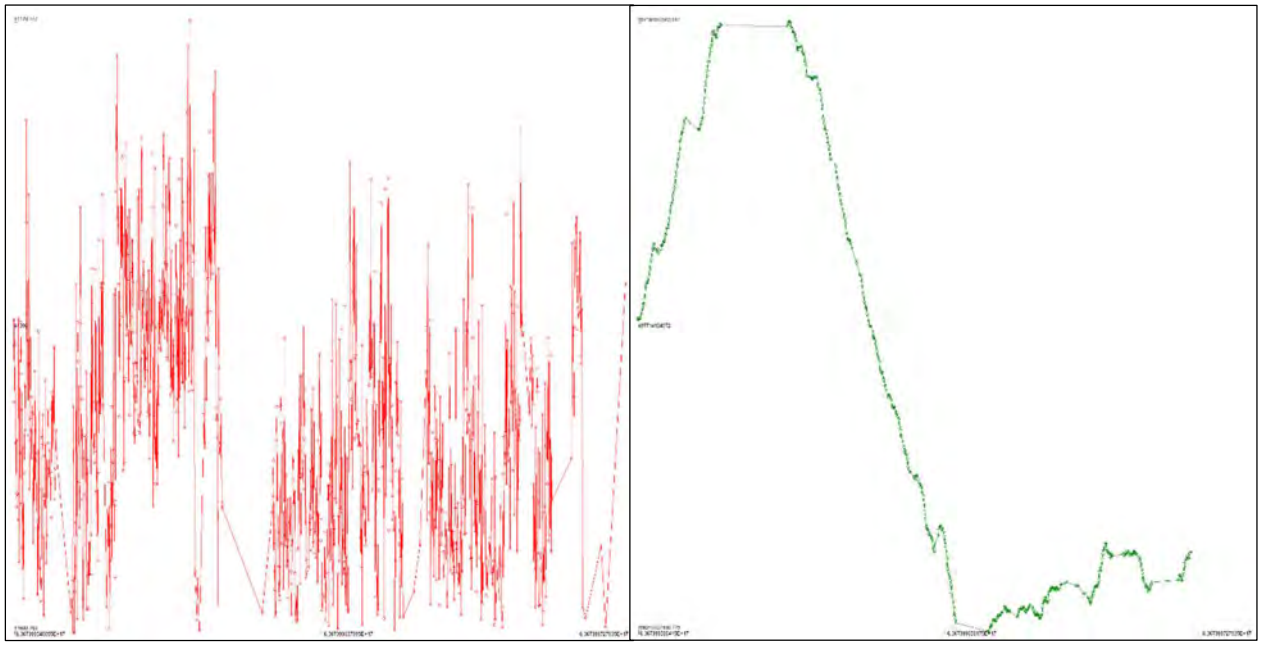


Figure 81 Same fragment, before (left) and after (right) application of the Gaussian smoothing. This allows to clearly distinguish a tendency that is almost imperceptible before treatment (Yann Donon A.).

after (right) gaussian filtering. The tendency to rise and the subsequent fall observable on the left image is practically imperceptible on the image before gaussian filtering.

In both proposed techniques, gaussian blurring is in the centre of the analysis, it allows through the adding a form of noise to understand a global picture instead of specificities. Gaussian blur can be adapted to n dimensions, in the case of time series, the filter should blur one dimension only, as such for σ the Gaussian distribution's standard deviation values are calculated as (Choe, 2014):

Equation 27 Gaussian transformation on unidimensional data

$$G(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2}{2\sigma^2}}.$$

After observations, a Gaussian kernel of length 100 and sigma $1 * 10^{-5}$ were selected for the filtering operation.

4.3.5 Thresholding

The blurring step significantly reduced the occurrence of false positive when analysing the data, however the general data volatility was a constant potential source of noise for the technique and the treatment of points of interest as described further in 4.4.1 Features comparison. It is the phase 3 on the algorithm's (Figure 3) and the technique's process (Figure 79). Average variance was a significant indicator of jittering periods. Not all periods presenting a significant variance presents jittering symptoms, but all periods of jittering presented a variance higher than average.

Thresholding allowed to highlight "proximity periods" as described above in 4.2.1 Label-related clustering. Concretely, the reference used for thresholding is all data presenting a variance lower than average over window of the 500 latest entries (approximatively 10 minutes on the data used). As for BIM this technique is used to define areas of interest into the data.

4.3.6 Filtering of areas of interest

Remining areas after Thresholding are selected as areas of interest if they retain their characteristics over a certain period. It is the algorithm's (Figure 3) and the technique's process (Figure 79) phase 4. The objective behind it is not to selects specifics events and their impact, which in the case of linear accelerator are common as explained earlier but to define changes

in tendencies, which can be indicators of an event. Detecting a tendency change early enough can as such approximate occurrence of significant events.

As for BIM, areas of interest are selected when they can be considered significant. As explained in this chapter, short events can be considered insignificant were showed uninformative, as such only jittering covering a period of 1500 entries or more (approximatively 30 minutes on the data used) were selected. Filtering large areas was not necessary as by the definition of the thresholding tool used, no area can cover more than 50% of the data samples.

Figure 82 illustrates the same data sample after blurring (Figure 82.1), Thresholding (Figure 82.2) and Filtering (Figure 82.3). The closer to red is an area, the higher is the average variance, meaning potential jittering. The area represented doesn't show any corresponding signal decay, however shades of red are present over the whole signal presented in Figure 82.1. After the Thresholding, only short periods over the peaks presented in the data remains potential areas of jittering according to SNiF. After filtering as described in this chapter, no area remains identified by the technique, which eliminates all possible false positive and allows to calculate and compare features only over data presenting significant chances of being symptomatic.

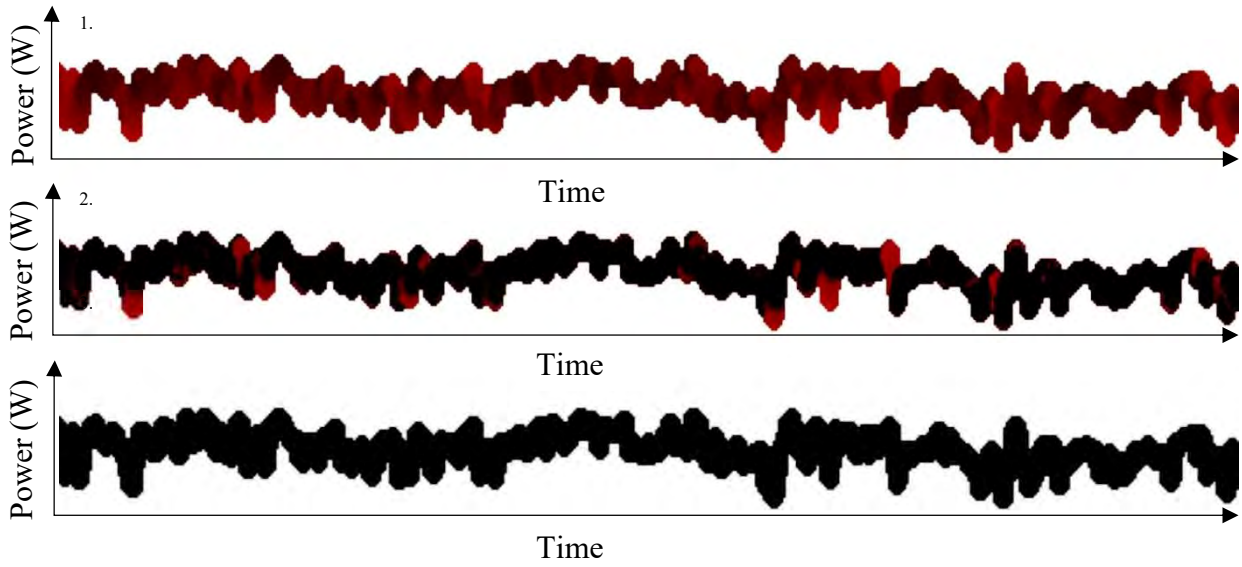


Figure 82 shows a data sample containing no jittering perids but rich in noise with 1. A data sample after blurring, red shades indicate likelyhood. 2. A data sample after thresholding and 3. a data sample after filtering. After the thirs stage, no false positive alter remains. The image colorimetry have been altered from the original content in order to highlight color differences.

4.3.7 Features selection

BIM featured two techniques for feature selection, both based on blobs. The nature of the data pushes to find other alternatives to delimitate features. It is represented in the algorithm's (Figure 3) and the technique's process (Figure 79) as phase 5. Jitters showed

different intensities, and length, the only constant observed so far is a change of points statistical distributions over periods. As the transitions observed from a point to another are continuous, a natural approach to estimate their distribution is Transition rate matrix (Syski, 1992), but using such matrix makes comparison as described in 4.4.1 challenging as it is then necessary to change the continuous matrix to a discrete ensemble. Gerschgorin circle theorem can be used in order to delimit frontiers, however those delimitation would be a source of potential noise differences in matrices comparison as they are defined according to each matrix individually (Gerschgorin, 1931).

Another alternative is the data transformation to a discrete ensemble prior to the matrix estimation. This alternative was retained for its simplicity of implementation and possibility to divide data using a uniform and statistically based approach. Data are clustered in 7 cluster of same vector size according to each value's variance and their distribution is reported in creating a stochastic matrix (Asmussen, 2003). Stochastic matrix describes Markov chains over a continuous space, the sum of all individual lines and columns is 1. As such elements for the Stochastic matrix P of size n by :

Equation 28 Stochastic matrix representation

$$P = \begin{matrix} P_{1,1} & \dots & P_{1,n} \\ \vdots & \ddots & \vdots \\ P_{n,1} & \dots & P_{nn} \end{matrix}.$$

Each element of P, $p_{i,j}$ can be calculated as :

Equation 29 Stochastic matrix element calculation

$$p_{i,j} = \Pr(X_{t+1} = x_j \mid X_t = x_i).$$

As for every line and column:

Equation 30 Stochastic matrix line and column summation

$$\sum_{j=1}^{S} P_{i,j} = 1 \ \& \ \sum_{i=1}^{S} P_{i,j} = 1.$$

The result of the operation is for each new data entry kept, after filtering, a Stochastic matrix is estimated in a window including the 1500 latest entries (approximatively 30 minutes on the data used), newly created matrix are then compared to each other as described below.

4.4 Results

As developped in 4.2, no technique was completely unsuccessful in recognizing and consistently isolate areas corresponding to beam decay. Performances, each technique allowed a better understanding of the data and their characteristics. Given the characteristics already analysed, SNiF success must be evaluated on the continuity of labelling through jittering periods, the main feature of success evaluation should as such be a reduced amount of false positive and negative compared to other periods. In this use case, false positive or negative impact is especially important as the technique is meant to condition choices in components replacement having an important financial impact.

The following chapters 4.4.1 Features comparison and 4.4.2 Features treatment, presents the steps taken to compare SNiF's functioning with other techniques and are not part the algorithm proposed (Figure 3). The results obtained are then described in this chapter.

4.4.1 Features comparison

As this stage the original data are reduced to sets of stochastic matrices, the comparison of such matrices distribution is a well know problem (Tu Grul Dayar, 2003) (Torgersen, 1991). This phase is beyond the task of key point detection and is used to compare the results obtained in the framework of the experiments presented, it is therefore not represented on the algorithm (Figure 3) but corresponds to technique's process (Figure 79) phase 6.1. The technique used differs again from BIM in terms of practice, due to the different nature of the data, however the concept of evaluating the general shape (or tendencies) of the sample is similar (Queen, 1967).

The stochastic matrices are extracted and clustered using k-means algorithm (Xiaoqian Wang, 2016). Least sum square is used extensively for such clustering for its simplicity (Løkse, 2014), it clusters n observations $(x_1, x_2, \dots, x_n)$ into k clusters S $(S_1, S_2, \dots, S_n)$, where k is given, respecting the condition $k\ (\leq\ n)$. The objective is to minimize the sum of squares by finding the following (Hans-Peter Kriegel, 2017), where μ is S_i 's mean of points:

Equation 31 Minimal sum square calculation

$$arg\ \min_{S} \sum_{i=1}^{k} \sum_{x \in S_i} ||\ x - \mu_i\ ||^2 =\ arg\ \min_{S} \sum_{i=1}^{k} |S_i\ |\ Var\ S_i.$$

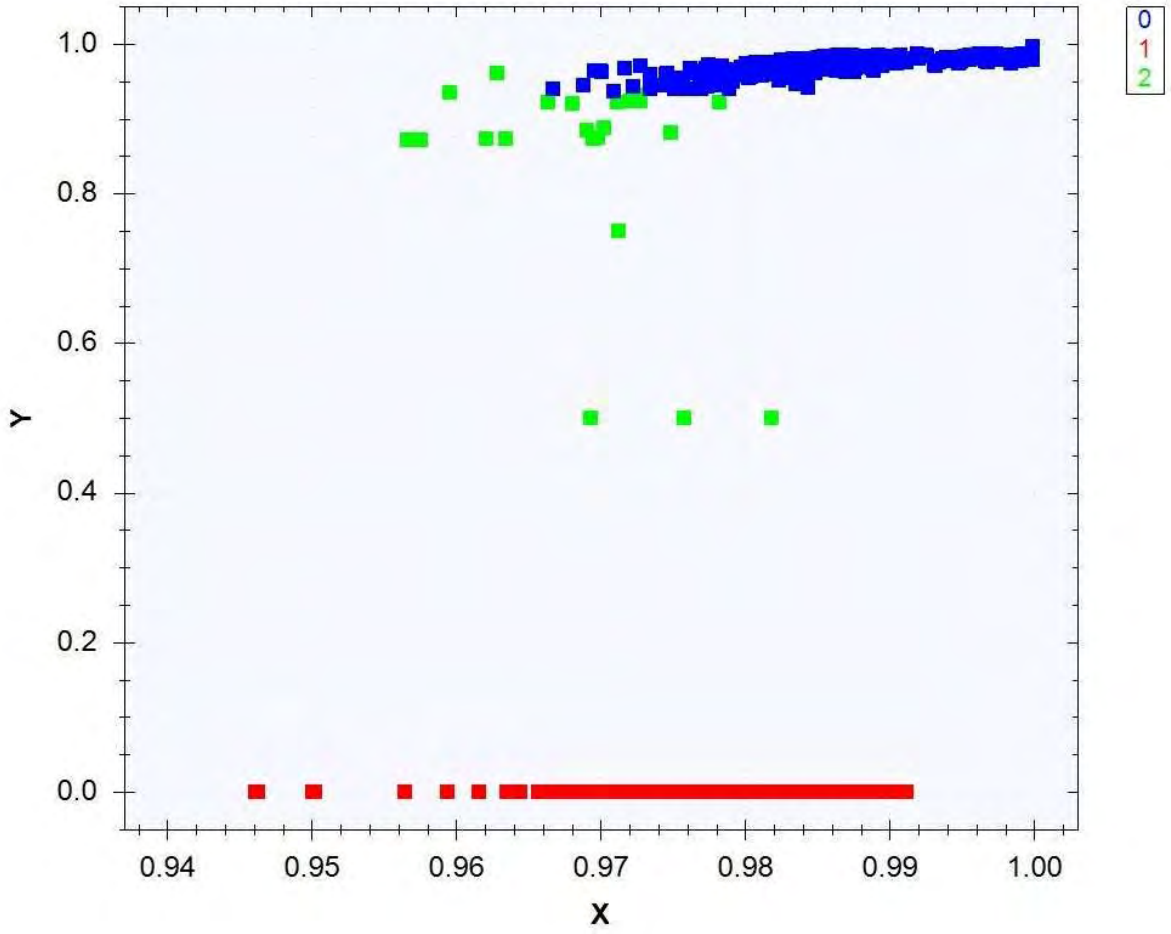


Figure 83 shows the clustering of a thousand windows stochastic matrices, with in blue regular operations, in green human interactions and in red periods of jittering.

In the case of the data observed, 3 clusters k are determined, one corresponds to regular operations, a second to periods of user interaction with the source and the third corresponds to periods of jittering. Figure 83 represents the clustering of a thousand windows reported on a million entries, processed with the Principal Component Analysis (PCA) procedure for visualisation (Jolliffe, 2002), in blue (Figure 83.0.), the periods corresponding to regular operations, in red (Figure 83.1.), the jittering periods and in green (Figure 83.2.), period corresponding to human interactions on the source. It can be noticed that even if human interactions provoke violent changes on the source, the data behaviour is close to regular operations period, those two being even difficult for the clustering technique to dissociate. However, periods of jittering have a very distinctive distribution, making it easy to separate them from the rest of the data and defining a strongly separated cluster.

In that context, it has been found that periods of jittering and their proximity areas as detailed 4.2.1 Label-related clustering corresponds to a change in the statistical distribution behaviour. This change operates with a growing variance intensity over a certain period before stabilizing as represented on Figure 84, where the behaviour mentioned earlier can be observed in green as the variance intensity augmentation slows down Figure 84.1. and almost stops Figure 84a.2. In other terms it signifies that periods of jittering presents early symptoms that are identical to the actual anomalies in SNiF's perception. This point is of importance as it allows prediction of course, but most importantly, it allows to approximate the moment an event leading to jittering occurs.

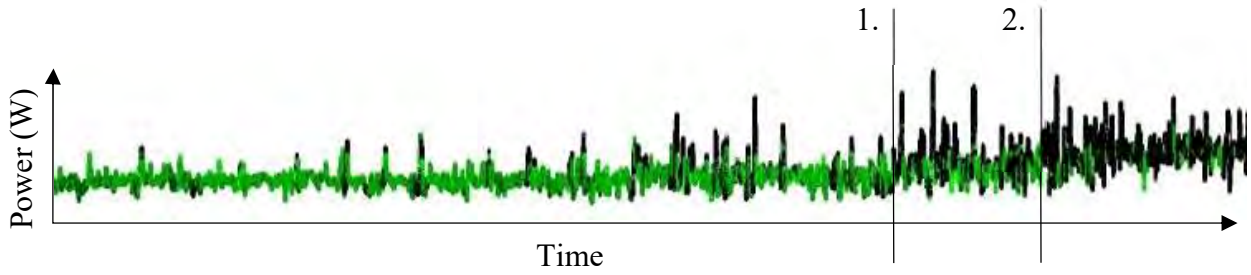


Figure 84 Shows a period of jittering, with the increase of variance intensity represented in green shades. Although the sample remains in a state compromising the beam quality, it is noticeable that the variance intensity amplification slows down (1) and almost stops (2).

The distribution of this feature is used to delimit proximity areas as used further for demonstration purpose; it is however not a feature used in the process of data featuring.

4.4.2 Features treatment

As the clusters are known, the operations described earlier are used successively for every new entry. Again, this step is part of the comparison process for the different experiments and is therefore not represented on the algorithm (Figure 3) but only in the technique's process (Figure 79) as phase 6.2.1 When an entry is classified into the cluster corresponding to jittering areas, alerts are thrown acknowledging the event with details of the event, the same way.

As several filters and windows acts on the entry before the output phase, every flag raised at this point is considered positive. Just like BIM created an image out of an homography matrix, SNiF matches features together and raises alerts accordingly.

4.4.3 Metrics

Precise metrics are difficult to evaluate in SNiF's case as, like for previous entries, the only available labelling in an expert approximation according to observable effect on the accelerator chains. But the problematics goes further as SNiF is successful in detecting

anomalies a significant time before their apparition using their proximity areas. Those areas were not perceptible by the domain expert, it is therefore not possible to compare the technique's success rate in proximity areas. This chapter therefore only compares BIM to other techniques on the areas of jittering detection, fragments detected by BIM as proximity areas are referred as "non evaluated" in Table 1. Moreover, all the false positive value found by SNIF were found at the border between the proximity area and the beginning of the jittering period as evaluated by the domain expert. This section of false positive is therefore not relevant as such and can be ignored as such. Compared to Table 1, Table 2 and Table 3, Table 1 is the only one not detecting false negative entries which is explainable when visualizing the corresponding data sample on Figure 86 and is developped below. SNiF labels between 210 and 8 times more points in anomaly periods depending on the technique it is compared to, due to the selection technique the technique uses, using large consequent windows, which creates a continuous and therefore consistent labelling through time.

	True positive	True negative	False positive	False negative	Non evaluated	Total
Absolute amount	249656	732859	18910	0	146086	1147511
Proportion of the total (%)	16.8	63.9	1.6	0	17.7	100

Table 4 represents the sequence analysis using SNiF technique results on a set of 2639281 entries. The "non evaluated" column refers to sections labelled as proximity areas by SNiF, those section can't be categorized differently as although they are representative, no form of labelling regarding them exists in the original dataset.

As referred above, Figure 86 mirrors the sample of data shown in Table 1, with in green labelling obtain from BIM, all categories of labelling together, the image highlights how proximity areas are detected by SNiF, which as mentioned earlier is formally a source of false positive alters. The image also shows how no segment of data besides the jittering area and its proximity area is labelled as potential positive.

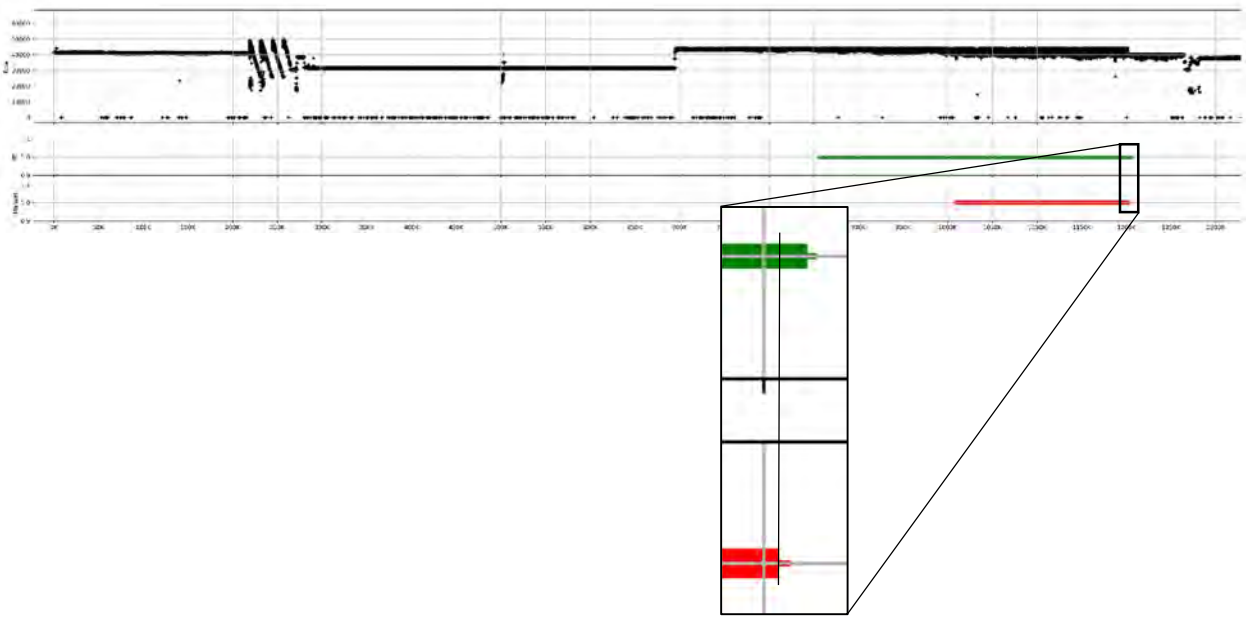


Figure 86 Illustration of the labelling obtained from a sample of LINAC 4's data with in black, the signal, in green the labelling obtain from SNiF and in in red labelling estimation by domain expert.

The fragment represented in Figure 86, in particular the jittering period and its proximity areas as defined in 4.4.1 is detailed in Figure 87, showing the proximity area in light read and periods of jittering in dark read as analysed by SNiF, those highlighting the areas marked as "Non evaluated" in Table 1.

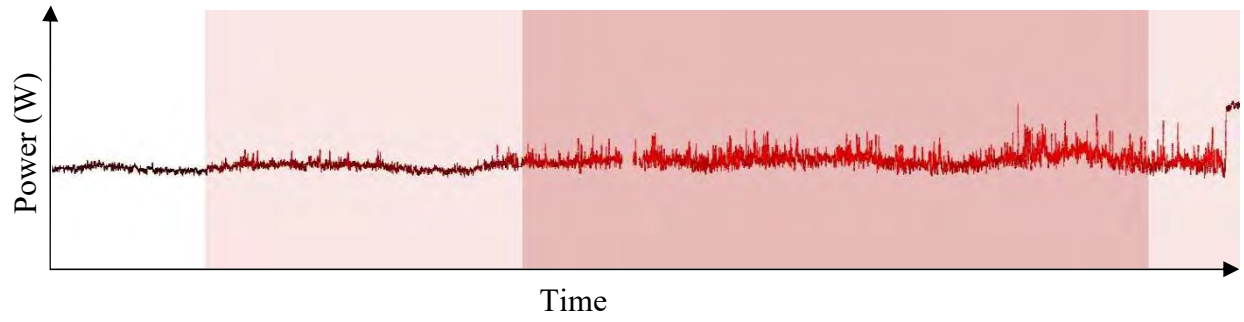


Figure 87 shows the fragment labelled as jittering in Figure 79 and Figure 80, with in light red periods labelled as proximity areas using SNiF.

4.4.4 Classification results

Overall, as seen on the previous chapter, results are difficult to evaluate in terms of quality from the moment a jittering period corresponding to a decay period in the beam quality has been isolated. It is possible to assess two points, which would be in the framework of the work developped:

- All jittering period has been consistently labelled
- No positive alert outside of jittering period or their proximity areas was found by the technique

In those aspects. BIM is a complete success, out of the LINAC4 datasets, extracted from the latest run, all the areas of anomalies marked by domain experts were labelled as such by the technique. SNiF consistently labelled those areas, keeping no false negative and no false positive outside of proximity areas, with proximity areas extending between 2 and 10 days before the jittering period.

4.5 Conclusion SNiF

4.5.1 Aims achieved

SNiF is a successful implementation of the algorithm proposed in this thesis, the technique fits the need expressed for the labelling of linear accelerator's captors time series, in particular for the recognition of jittering areas. It was the only tested technique succeeding to label the LINAC's captors time series consequently and with a minimal amount of false classifications. Consequently, the technique will be implemented in production from 2020 in CERN's accelerator complex. It doesn't complete the SmartLINAC project mentioned earlier but SNiF is a viable solution to a first and essential step towards it.

4.5.2 Tasks solved

Altogether, SNiF showed, like BIM, good and unique performances in selecting large features in time series. The technique detects tendencies on long periods, it is therefore adapted to tasks as detecting change of behaviour over time over specific and punctual features, which would risk being ignored. The unavailability of large data samples allowing a relevant comparison between technique as was conducted with BIM explain why only a reduced amount of metrics were presented to establish BIM performances. The metrics presented are however enough to assess the technique viability and highlight its specific mode of functioning as:

- The technique successfully identified all the areas of jittering identified by domain experts on LINAC4's datasets.
- The technique presented no false positive labelling outside of the jittering periods' proximities areas.
- The technique presented no false negative labelling through the data analysed.

4.5.3 Statement

SNiF, is an implementation of the algorithm presented in this thesis and was compared to existing solutions following a scientific methodology. The experiments performed shows the technique relevance for feature selection in noised data. The technique allows detecting features with a high consistency and confidence when compared to existing techniques, proving SNiF's relevance when applied to noised images.

4.5.4 Final word on SNiF

The approach and the size of features selected makes the technique especially relevant for the selection of features in the shape of change of behaviour in the data. As such, the

technique is implemented in the engineering in SmartLINAC process, making a step towards the development of a functioning platform for maintenance planning in linear accelerators.

Like BIM, SNiF uses existing and implemented techniques for data processing, making its implementation accessible, which relevant strength of the algorithm presented in this thesis and its subsequent techniques.

5 Conclusion

5.1 Aims achieved

The algorithm presented in this thesis is proposed as a solution for features detection on noised data. It reached the objectives advanced in the introduction in terms of metrics, moreover its approach showed specific characteristics do not present in other features detection technique, whether on images or time series.

Out of the two implementations of the algorithm BIM technique showed:

- To be the fastest tested technique by an average of 34% to the second fastest.
- The features with the highest quality, unlike other tested techniques, the chances of BIM features being correct is higher average (66%).
- To be second by 0.4% for image stitching on regular sets with a success rate of 93.8%.
- To register the most successes when tested on noised datasets, with a success rate of 65%. Almost three times superior then the second-best performing technique.

The second algorithm's, SNiF, implementation showed:

- A unique feature selection approach, selecting consistent windows instead of points.
- Few to no false negative labelling due its approach.
- Few to no false positive labelling influenced by the ambient noise.
- The ability to detect periods of change of the signal's statistical repartition from early symptoms.

The success enounced above confirms the algorithm's legitimacy to figure among feature detection solutions, in particular on noised data and complete the aims enounced in this thesis.

5.2 Tasks solved

The objectives enounced in 1 Aim were the development of a technique allowing key points identification on noised images, in order to demonstrate the algorithm described in Figure 3. By the standards developed in this thesis, the tasks enounced in order to demonstrate the algorithm's performance were solved as:

- The technique developped, BIM, showed a success rate of 65%, outclassing by more than twice the success rate registered by the most performant technique of the testing pool and reaching from far the objective of 46%.
- It was showed to be a polyvalent technique, being showing itself more robust than other testing technique with all sources of noise tested.
- The technique showed itself faster than other technique for stitching operations
- BIM uses less points and of better quality than any other tested technique
- Unlike common approach, the technique doesn't estimate missing information (Zhang, 2015) due to noise but consider noises as part of the data, which could alter the data and mislead stitching techniques

Those statements confirm the relevancy of the technique when dealing with noised data in the form of images.

The second objective enounced in 1 Aim was the development of a technique allowing key points detection and comparison on noised time series using the same algorithm described in Figure 3, resulting in SNiF showing great performances and unique characteristics relative to the algorithm. Again, as described in the thesis, the technique developed was a success according to criteria defined as:

- SNiF identified 100% of jitters in CERN's LINAC4 datasets, also detecting proximity areas, allowing to foresee beam decay periods.
- The technique showed excellent metrics, not detecting false positive alerts outside of the jittering periods or their proximity area and presenting a continuous labelling of noised series, thus not raising false negative alerts.
- SNiF performs good in identifying significant periods presenting changing statistical characteristics.
- The technique labelling is consistent over long periods of time, thus producing very few false negative alerts.

The goal enounced in this thesis was the proof of concept for the algorithm enounced in Figure 3 the development of techniques (BIM and SNiF) was enounced as a demonstration of the algorithm's functioning. It is therefore necessary to compare the algorithm enounced in the figure with those used for BIM and SNiF to assess their conformity. Figure 88 compares both BIM and SNiF process, in the figure, red represent steps that depends on the data type, blue steps corresponding to the algorithm presented in this thesis and white steps depending only on the comparison approach used to assess the techniques in this article. For both techniques it is observable that the red steps depend on the data category that was used for comparison. For BIM, those steps correspond to minor corrections in images preparing images for their processing. For SNiF, it includes non-informative features filtering, which in the case of the data obtained from LINAC4 is a very specific category of noise (measurement inconsistency). Those steps can be considered therefore as a preparation for a treatment using the algorithm proposed in the thesis. Sections in white are irrelevant here as they serve to compare implementations and are not part of the processes themselves for other reasons. Finally, blue parts, representing steps proposed in the thesis are predominant, both techniques use all steps proposed in the same order, with the same input parameters needs, showing their conformity to the algorithm proposed. The similarities between the two techniques and extra step being only needed as a way to prepare data for the central algorithm, shows that both BIM and SNiF are indeed based on the algorithm proposed in this thesis and that this algorithm is in the core of both techniques' functioning.

Figure 88's blue steps, highlight the contributions proposed in this thesis as they are the core of the algorithm proposed.

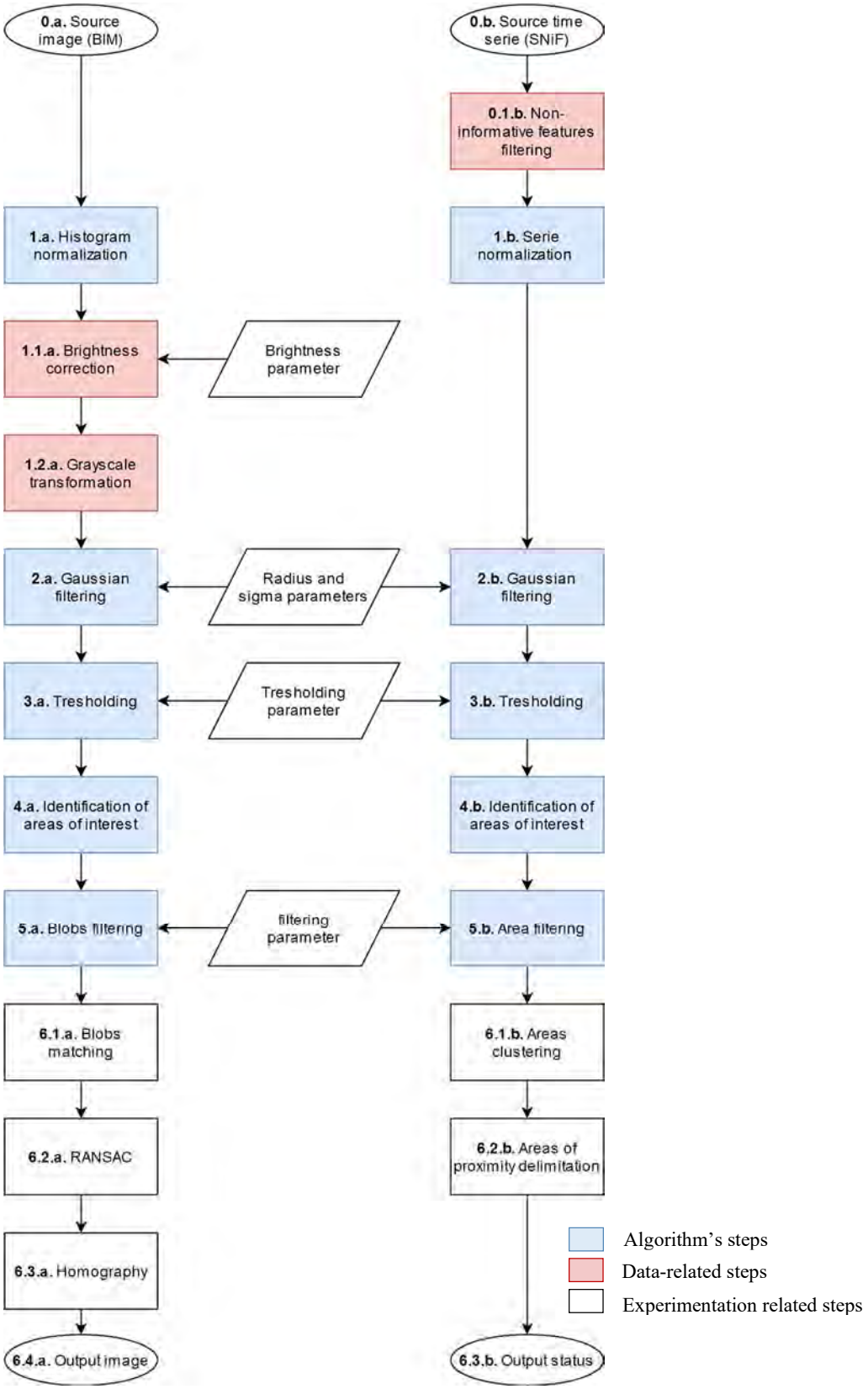


Figure 88 BIM and SNiF process comparison with similar steps (blue) steps depending on the data source (red) and step relative to the experimentation (white).

5.2.1 Outside's scope evolutions

From the conclusions drawn in the previous chapter, a relevant next step would be to keep comparing the technique on noised series with more dimensions such as hyperspectral analysis in order to obtain one more dimension than images, however such research will not be conducted in the framework of this thesis, if a relevant and expressed need of such technique appears then the development will be considered.

Another relevant point to explore is parameter selection using machine learning approaches. In this thesis, parameters are given according to research done on the techniques which corresponds to the approach used by the other techniques used to assess both techniques. However, better results could potentially be obtained by selecting parameters using neural network to select parameters relatively to every sample of data, thus making both techniques "of fit". Moreover, this step will be a necessity to conduct the SmartLINAC project and will therefore be introduced in the project's next steps as SNiF will have to adapt, without supervision to many different captors for different linear accelerators from both scientific and medical sources. This step was excluded from the framework of this thesis as although it highlights again the similarity of both techniques around the main algorithm, it is not relevant as part of the algorithm described in the thesis.

The challenge mentioned above could be tackled for images using a generated dataset including numerous images of different kind and presenting a variety of noises with different intensities. This dataset would then be treated by BIM using a variety of different parameters and calculating the precision of result obtained for every set of parameters before feeding to a neural network sources images and the most successful sets of parameters.

Time series parameter selection could be achieved by running SNiF on new datasets on an initialization run while labelling incoming data based on default parameters, feeding to a neural network the labelled data in order to compare them with similar labels from already treated datasets.

Both techniques have been approached, assessing their feasibility, but the research being in a preliminary state, it is yet too early to draw conclusions over its functioning, moreover, as described above it falls outside of this thesis's scope.

5.3 Statement

The work presented in this thesis is novel and scientifically relevant; the algorithm proposed performed the series of tasks enounced to assess its performances successfully. The

algorithm's implementations are used by the industry, in particular at CERN. they allowed to solve problems so far unanswered. The algorithm allows solving feature detection in noised with higher performances on the tested scope than existing techniques. This thesis also confirms the experiments performed reflects a scientific methodology proving the algorithm's legitimacy to offer a novel way to select features in data, in particular noised, through its unique characteristics and performances.

5.4 Final word

This thesis proposes a new approach that has not for pretention to be a breakdown in terms of science but offers a new perspective in terms of problem solving, in particular in the case of noised data. If none of the techniques shows new elements, they use existing elements in a way that had not been approached before and it results in a working solution for noised data, with unique metrics when compared to other technique.

The centre of the work presented in this paper is an algorithm which, although not containing elements of novelty in its workflow, is a novelty in terms of both approach and resulting algorithm. Its core simplicity makes its implementation very approachable which apart from facilitating its use of a relevant feature when the question of solutions implementation quality grows rising in science (Kriegel, 2017).

The algorithm presented kept through both techniques its integrity in terms of simplicity and source of inspiration. As a final word, as simple as it may sound, the techniques derived from the algorithm functions and are successfully used to solve issues that so far would not find answers thus making the work presented in this thesis a scientific and engineering success.

Table of figures

Table of tables

Table of equations

Bibliography

Accord.net. (2019, December 05). *Fast Retina Keypoint (FREAK) detector.* . Retrieved from Accord.NET Framework: http://accord-framework.net/docs/html/T_Accord_Imaging_FastRetinaKeypointDetector.htm

Accord.net. (2019, December 10). *Grayscale image using BT709 algorithm.* Retrieved from Accord.net framework: http://accord-framework.net/docs/html/F_Accord_Imaging_Filters_Grayscale_CommonAlgorithms_BT709.htm

Accord.net. (2019, December 05). *Speeded-up Robust Features (SURF) detector*. Retrieved from Accord.NET Framework: http://accord-framework.net/docs/html/T_Accord_Imaging_SpeededUpRobustFeaturesDetector.htm

Accord.net. (2019, December 05). *Harris Corners Detector*. Retrieved from Accord.NET Framework: http://accord-framework.net/docs/html/T_Accord_Imaging_HarrisCornersDetector.htm

Alexandre Alahi, R. O. (2012). FREAK: Fast retina keypoint. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (pp. 510-517).

Arabie, L. H. (1985). Comparing partitions. *Journal of Classification, 2*(1), 193–218.

Asmussen, S. R. (2003). Applied Probability and Queues. In *Stochastic Modelling and Applied Probability* (pp. 3-4). New York: Springer.

Au, A. (2013). *Development of Multiview Image/Video Stitching Systems for Mobile Devices.* British Columbia: Simon Fraser University.

B. S. Everitt, A. S. (1998). *The Cambridge Dictionary of Statistics.* Cambridge, UK New York: Cambridge University Press.

Bhattacharyya, A. (1943). On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society. 35*, 99–109.

Boncelet, C. (2009). Image Noise Models, Salt and Pepper Noise. In CharlesBoncelet, *The Essential Guide to Image Processing (Second Edition)* (pp. 143-167). Austin: Elsevier Inc.

Brown, M. A. (2005). *Multi-Image Matchingusing Invariant Features.* Vancouver: University of British Columbia.

Brown, R. G. (1992). *Introduction to random signals and applied Kalman filtering* (Vol. 3). New York: Wiley.

C., K. (2011). Optimization approach to adapt Kalman filters for the real-time application of accelerometer and gyroscope signals' filtering. *Digital Signal Processing, 21*(1), 131-140.

Carlos Lopez-Martinez, E. P. (2007). On the Extension of Multidimensional Speckle Noise Model From Single-Look to Multilook SAR Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 305-320.

CERN. (2018). *Key Facts and Figures – CERN Data Centre.* Geneva: CERN IT department.

CERN. (2019, July 29). *The CERN accelerator complex.* Retrieved from CERN Document Server: https://cds.cern.ch/record/2684277

Chan, M. (1996). Optimal Output-Sensitive Convex Hull Algorithms in Two and Three Dimensions. *Discrete & computational Geometry*, 361-368.

Choe, C. W.-Y. (2014, June 30). Two-way partitioning of a recursive Gaussian filter in CUDA. *EURASIP Journal on Image and Video Processing*, pp. 1-12.

Chris Harris, M. S. (1988). A combined corner and edge detector. *Proceedings of the Alvey Vision Conference*, (pp. 147–151). United Kingdom.

David Pistenmaa, N. C. (2017). Developing medical linacs for challenging regions. *CERN Courier*(March).

Di Meglio, A. (2017). *Facing up to the exabyte era.* Geneva: CERN courier.

Diebold, F. (2007). *Elements of Forecasting (Fourth ed.).* Philadelphia: University of Pennsylvania.

Dubrofsky, E. (2009). *Homography Estimation.* Vancouver: Carleton University.

Evans, C. (2009). *Notes on the OpenSURF Library.*

Forouzanfar, M. A.-M. (2007). Speckle reduction in medical ultrasound images using a new multiscale bivariate Bayesian MMSE-based method. *IEEE 15th Signal Processing and Communication Applications Conf.*, 1-4.

Freud, S. (1912). *Recommendations for Physicians Practicing Psychoanalysis.* London: Hogarth Press and the Institute of Psycho-analysis.

Gary Bradski, A. K. (2008). *Learning OpenCV.* Sebastopol, CA: O'Reilly.

Gerschgorin, S. A. (1931). Uber die Abgrenzungder Eigenwerte einer Matrix. *Bulletin de l'académie des sciences de l'URSS*, 749-754.

Graham, R. (1972). An Efficient Algorithm for Determining the Convex Hull of a Finite Planar Set. *Information Processing Letters*, 132-133.

Grubbs, F. E. (1950). Sample criteria for testing outlying observations. *Annals of Mathematical Statistics*, 27-58.

Grubbs, F. E. (1969). Procedures for Detecting Outlying Observations in Samples. *Technometrics*, 1–21.

Hans-Peter Kriegel, E. S. (2017). The (black) art of runtime evaluation: Are we comparing algorithms or implementations? *Knowledge and Information Systems*, 341–378.

Hantao Yao, S. Z. (2016). Coarse-to-Fine Description for Fine-Grained Visual Categorization. *IEEE Transactions on Image Processing*, 4858 - 4872.

Herbert Bay, T. T. (2006). SURF: Speeded Up Robust Features. In B. H. Leonardis A., *Lecture Notes in Computer Science* (Vol. 3951, pp. 404-417). Berlin, Heidelberg: Springer.

International Telecommunication Union. (2015). *Recommendation ITU-R BT.709-6.* Geneva: ITU.

Jan Hosang, R. B. (2017). *Learning non-maximum suppression.* Saarbrücken, Germany: Max Planck Institut für Informatik.

Jarvis, R. (1973). On the identification of the convex hull of a finite set of points in the plane. *Information Processing Letters*, 18-21.

Jin-Sheng Guf, W.-S. J. (1996). The Haar wavelets operational matrix of integration. *International Journal of Systems Science* , 623-628.

Johan Nysjö, A. H. (2013). Optimal RANSAC - Towards a Repeatable Algorithm forFinding the Optimal Set. *Journal of WSCG*, 21-30.

Jolliffe, I. T. (2002). *Principal Component Analysis.* New York: Springer Series in Statistics.

Jong-Sen Lee, E. P. (2009). Polarimetric SAR speckle filtering. *Optical science and engineering series*, 142.

Krarup, J. (2018, June 19). *ImageComparison 2.0.4*. (Microsoft nuget) Retrieved March 04, 2019, from https://www.nuget.org/packages/ImageComparison/

Kreyszig, E. (1979). Advanced Engineering Mathematics. Wiley.

Kriegel, H. S. (2017). The (black) art of runtime evaluation: Are we comparing algorithms or implementations? *Knowledge information systems*, 341–378.

Kriegman, D. (2007). Homography Estimation. *Computer Vision I*, 1-3.

Lim, K. (2016). Fading Kalman filter-based real-time state of charge estimation in LiFePO4 battery-powered electric vehicles. *Applied Energy*(2016), 40-48.

Løkse, S. (2014). *Joint ranking and clustering based on Markov Chain transition probabilities learned from data.* Tromsø: The Arctic University of Norway, Department of Physics and Technology.

Lowe, D. G. (2004). Distinctive Image Featuresfrom Scale-Invariant Keypoints. *International Journal of Computer Vision*, 1-28.

Lu Y, H. Z. (2018). Multiperspective image stitching and regularization via hybrid structure warping. *Computing in Science and Engineering, 20*(2), 10-23.

Martin A. Fischler, R. C. (1980). *Random Sample Consensus: A Paradigm for Model Fitting withApplications to Image Analysis and Automated Cartography.* Menlo Park: SRI international.

Mayur Datar, A. G. (2012). Maintaining Stream Statistics over Sliding Windows. *SIAM Journal on Computing, 31*(6), 1794–1813.

Mount, D. M. (2012). *CMSC 754 Computational Geometry.* College Park: Department of Computer Science, University of Maryland.

Nan Li, Y. X. (2018). Quasi-Homography Warps in Image Stitching. *IEEE TRANSACTIONS ON MULTIMEDIA, 20*(6), 1365- 1375.

National Telecommunications and Information Administration. (1949). *Federal Standard 1037C.* General Services Administration (USA).

O. Rey Orozco, A. A.-E. (2018). PERFORMANCE EVALUATION OF LINAC4 DURING THE RELIABILITY RUN. *9th International Particle Accelerator Conference* (pp. 1-4). Vancouver: JACoW Publishing.

OpenCV. (2014). *Image Filtering.* Retrieved from OpenCV documentation: https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=gaussianblur#gaussianblur

OpenCV. (2019 , December 5). *Features detection and description*. Retrieved from OpenCV-Python Tutorials: https://docs.opencv.org/3.4/db/d27/tutorial_py_table_of_contents_feature2d.html

OpenCV. (2020, March 3). *Geometric Image Transformations*. Retrieved from OpenCV documentation: https://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html

OpenCV. (3, October 2019). *Image filtering, getGaussianKernel() function*. (OpenCV) Retrieved October 2019, 3, from https://docs.opencv.org/master/d4/d86/group__imgproc__filter.html#gac05a120c1ae92a6060dd0db190a61afa

Parzen, E. (1962). On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics, 33*(3), 1065–1076.

Pierre Donon, R. P. (n.d.). Key point detection on images, an improved version of BIM.

Pooja Ghosh, A. P. (2015). Comparison of Different Feature Detection Techniques for Image Mosaicing. *ACCENTS Transactions on Image Processing and Computer Vision*.

Queen, J. M. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, 281–297.

Rafael C. Gonzales, R. E. (2007). *Digital Image Processing.* Upper Saddle River: Prentice Hall.

Rahul Raguram, J.-M. F. (2009). Exploiting Uncertainty in Random Sample Consensus. *12th International Conference on Computer Vision* (pp. 2074-2081). Kyoto: IEEE.

Ribeiro, M. I. (2004). *Gaussian Probability Density Functions:Properties and Error Characterization.* Lisbon: Instituto Superior Tcnico.

Ronald Graham, F. Y. (1983). Finding the convex hull of a simple polygon. *Journal of Algorithms*, 324-331.

Rosenblatt, M. (1956). Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics, 27*(3), 832–837.

Rustam Paringer, Y. D. (2020). *Модификация метода сопоставления размытых изображений.*

S., G. M. (2011). *Kalman Filtering.* Heidelberg: Springer.

Salkind, N. (2010). *Encyclopedia of Research Design.* Thousand Oaks: Sage.

Samara National Reasearch University. (2018 , October 22). *Samara University teams up with the European Organization for Nuclear Research through CERN openlab*. (Samara National Reasearch University) Retrieved December 03, 2019, from https://ssau.ru/english/news/15950-samara-university-teams-up-with-the-european-organization-for-nuclear-research-through-cern-openlab

Sarabjeet Kaur, E. S. (2017). Analysis of Image Stitching for Noisy Images using SIFT. *International Journal of Advanced Research in Computer Science, 8*(5), 2078-2082.

Sergey Bezryadin, P. B. (2007). Brightness Calculation in Digital Image Processing. *International Symposium on Technologies for Digital Photo Fulfillment.* San Francisco.

Shapiro Linda, S. G. (2001). *Computer vision.* Prentice Hall: Pearson.

Shumway R.H., S. D. (2017). *Time series analysis and its applications: with R examples.* New York: Springer.

Souza, C. d. (2014). *Automatic Image Stitching with Accord.net.* December.

Sue, M. (1981). *Radio frequency interference at the geostationary orbit.* Pasadena: NASA Jet Propulsion laboratory.

Syski, R. (1992). *Passage Times for Markov Chains.* IOS Press.

Teh, C.-H. C. (1989). On the detection of dominant points on digital curves. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 859-872.

Texas Instruments. (2007). *Noise analysis in operational amplifier circuits.* Digital Signal Processing Solutions.

Thomas H. Cormen, C. E. (2009). . *Introduction to Algorithms,.* Third Edition. MIT Press.

Thomas H. Lee, A. H. (2000). Oscillator Phase Noise: A Tutorial. *IEEE JOURNAL OF SOLID-STATE CIRCUITS*, 326-336.

Torgersen, E. (1991). *Stochastic orders and comparison of experiments.* Oslo: University of Oslo.

Tu Grul Dayar, J.-M. F. (2003). Transforming stochastic matrices for stochastic comparison with the st-order. *RAIRO Operations Research*, 85-97.

Vic Barnett, T. L. (1994). *Outliers in Statistical Data, 3rd Edition.* Wiley.

Weisstein, E. W. (2019, December 10). *Gaussian Function*. Retrieved from Wolfram MathWorld: http://mathworld.wolfram.com/GaussianFunction.html

Xiaoqian Wang, F. N. (2016). *Structured Doubly Stochastic Matrix forGraph Based Clustering.* San Francisco: ACM.

Y Donon et al. (2019). Blur-robust image registration and stitching. *Journal of Physics: Conference Series*, 1-11.

Yanfang Li, Y. W. (2008). Automatic Image Stitching Using SIFT. *International Conference on Audio, Language and Image Processing.* Shanghai.

Yann Donon, A. K. (2019). ANOMALY DETECTION AND BREAKDOWN PREDICTION IN RF POWER SOURCE OUTPUT: A REVIEW OF APPROACHES. *Proceedings of*

the 27th International Symposium Nuclear Electronics and Computing (NEC'2019) (pp. 99-104). Budva: CEUR-WS.

Yann Donon, A. K. (2020). Extended anomaly detection and breakdown prediction in LINAC 4's RF power source output. *ITNT 2020 proceedings*.

Yann Donon, R. P. (2020). Brightness normalization for Blurred Image Matching. *TBP*.

Yann Donon, R. P. (2020). Parameters selection for Blurred Image Matching. *TBP*.

Zaragoza J, C. T.-J.-H. (2014). As-projective-as-possible image stitching with moving DLT. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(7), 1285-1298.

Zhang, Z. (2015). *Image Noise: Detection, Measurement, andRemoval Techniques.* Knoxville: Department of Electrical Engineering and Computer Science.