HAL
open science

From Large Time Series to Patterns Movies: Application to Airbus Helicopters Flight Data

Benjamin Chazelle, Pierre-Loic Maisonneuve, Ammar Mechouche, Jean-Marc Petit, Vasile-Marian Scuturici

To cite this version:

Benjamin Chazelle, Pierre-Loic Maisonneuve, Ammar Mechouche, Jean-Marc Petit, Vasile-Marian Scuturici. From Large Time Series to Patterns Movies: Application to Airbus Helicopters Flight Data. 25th European Conference on Advances in Databases and Information Systems (ADBIS 2021), Aug 2021, Tartu, Estonia. pp.213-226, 10.1007/978-3-030-82472-3_16 . hal-03325749

HAL Id: hal-03325749

https://hal.science/hal-03325749v1

Submitted on 25 Aug 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

From Large Time Series to Patterns Movies: Application to Airbus Helicopters Flight Data

Benjamin Chazelle[1] Pierre-Loic Maisonneuve[2]
Ammar Mechouche[2] Jean-Marc Petit[1]
Vasile-Marian Scuturici[1]
[1] Universit de Lyon, INSA Lyon, CNRS, UCBL, LIRIS, Villeurbanne, France
{jean-marc.petit,marian.scuturici}@liris.cnrs.fr
[2]Airbus Helicopters, Marignane, France
{ammar.mechouche, pierre-loic.maisonneuve}@airbus.com

August 25, 2021

Abstract

Huge amount of multivariate time series (TS) data are recorded by helicopters in operation, such as oil temperature, oil pressure, altitude, rotor speed to mention a few. Despite the effort deployed by Airbus Helicopters towards an effective use of those TS data, getting meaningful and intuitive representations of them is a never ending process, especially for domain experts who have a limited time budget to get the main insights delivered by data scientists.

In this paper, we introduce a simple yet powerful and scalable technique for visualizing large amount of TS data through *patterns movies*. We borrow the co-occurrence matrix concept from image processing, to create 2D pictures, seen as *patterns*, from any multivariate TS according to two dimensions over a given period of time. The cascade of such patterns over time produces so-called *patterns movies*, offering in a few seconds a visualisation of helicopter' parameters in operation over a long period of time, typically one year.

We have implemented and conducted experiments on Airbus Helicopters flight data. First outcomes of domain experts on patterns movies are presented.

1 Introduction

For safety and maintenance reasons, many physical sensors have been installed on operating helicopters. From a data perspective, the Flight Data Continuous Recorder (FDCR) collects Time Series (TS) from physical sensors of the machine, usually at a frequency of 2 hertz (Hz). Over the last decade, Airbus Helicopters have gathered data on hundreds of thousands flight hours, over

hundreds of helicopters operated by different customers worldwide, on many different types of missions. To face with such huge amount of TS data, a Big Data platform has been deployed to enable the storing and processing capabilities, offering new opportunities to domain experts, especially for troubleshooting and predictive maintenance [7]. Time series analysis is still an active research domain, see for example [2], [1], [3] or [10] for a survey. Despite the effort deployed by Airbus Helicopters towards an effective use of those TS data, getting meaningful and intuitive representations of them remains a never ending process.

In the past, data were mainly used for troubleshooting purposes, for instance to understand the conditions triggering an unexpected incident. The process was limited to the exploitation of data from the flight where the incident occurred or the flight before, analyzed using classical TS visualization softwares such as Grafana[1], Kibana[2] or in-house dedicated tools.

For years, data collected from helicopters are also used to better understand the real usage spectrum of the different helicopters sub-systems (such as lubrication system, starter generator, hydraulic pump, landing gear...). Understanding the behaviour of these sub-systems in different contexts allows us to optimize their future (re)-design.

Now, more and more efforts are made to develop predictive maintenance capabilities for helicopters systems, based on the whole in-service data made available. However, this is a more challenging topic which requires a strong involvement of System Design Responsible (SDR) experts who have a deep knowledge of their respective systems.

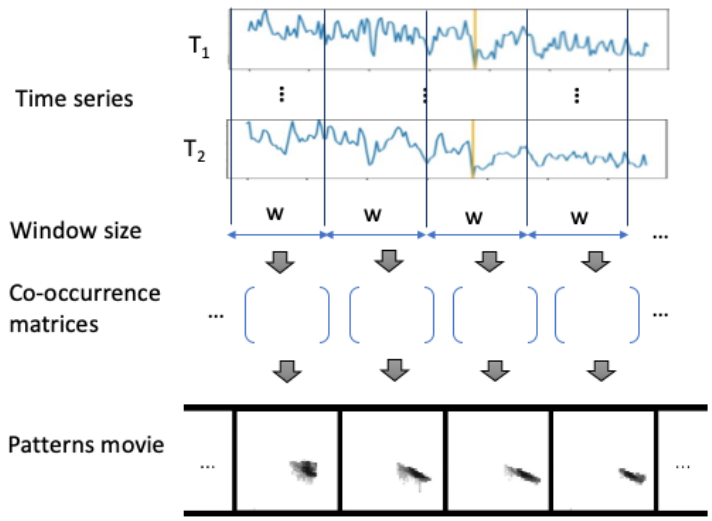
Predictive maintenance algorithms consist in general of monitoring "quantities" that should respect some conditions which can be considered as the normal operating behaviour of the monitored system. Then when this quantity no longer respects the conditions, an alert is raised. Also, often, such conditions come from hypothesis formulated by SDR, and consist in general of correlations that should be preserved over time between certain flight parameters under certain flight conditions, such as time between pilot actions and systems reactions, correlations between systems temperatures and pressures etc. Quick testing / verication / validation of SDR hypothesis is then very important for their efficient involvement in the predictive maintenance development. Nevertheless, SDR do not have time neither necessarily the required data science skills to mine, quickly and autonomously, massive collected TS data. Thus, it is quite important to provide them with relevant and adequate artefacts that allow them to get a simplied access and analysis of their TS data.

Turning large TS into useful knowledge for domain experts is clearly not an easy task at all. In this paper, we introduce a simple yet powerful and scalable technique for visualizing large TS data through patterns movies.

An overview of our approach is given in Figure 1. The basic idea relies on the visualization of correlations between two (flight) parameters over a large period of time. This large period is split into non-overlapping time windows.

[1] `https://grafana.com/`
[2] `https://www.elastic.co/fr/kibana`

Figure 1: Sketch of patterns movie construction



For each time window we build one image (or pattern), corresponding to the co-occurrence matrix of the two parameters, aggregating the TS information from this time window. By assembling successive images, one for each time window, we obtain a so-called "patterns movie".

The resulting movie points out how patterns slowly evolve over time. Domain experts have the opportunity to visualize in a short period of time – typically less than a minute – millions of records and observe trends related to helicopter usage.

We have implemented and conducted experiments on Airbus Helicopters flight data. First outcomes of domain experts on patterns movies are presented.

To the best of our knowledge, the use of co-occurence matrix for aggregating large TS data and their visualization with patterns movies is a new contribution which has many advantages:

- Pattern movies are very convenient for domain experts to better understand their TS data
- The proposed process turns out to be scalable, almost linear in the size of the TS data.

2 From TS data to patterns videos

Let (T_1, T_2) be two numerical TS variables over the same period of time $\mathcal{T}$ and $w_1, w_2, \ldots, w_n$ a succession of non-overlapping time windows of the same size, with $w_i << \mathcal{T}$ for each $i \in \{1, \ldots, n\}$.

We denote by $\hat{T_1}$ and $\hat{T_2}$ a discretization of these TS. Many techniques could be applied such as equal-width discretization or equal-frequency discretization. Furthermore, external knowledge provided by experts should be taken into account. Details are omitted.

For a time window w, we denote by T_1^w, T_2^w the part of (T_1, T_2) that fall into w.

The proposed process applied on every time window w is as follows:

1. Discretize T_1^w and T_2^w into $\hat{T_1^w}$ and $\hat{T_2^w}$
2. Count the number of occurrences of any pairs of values from $(\hat{T_1^w}, \hat{T_2^w})$.
3. Build the co-occurence matrix $\hat{M}_w$
4. Generate a picture associated to $\hat{M}_w$
5. Integrate the picture into an MPEG file, i.e. the movie file is composed by successive frames, one frame for each time window $w_1, \ldots, w_n$

We reuse classical notions of co-occurrence matrix from image processing, useful in texture analysis of 2D images [4]. In our case, we consider that each co-occurence matrix is a simple image which captures useful information (*patterns*) about the process of interest. The co-occurrence matrix can be seen also as a multidimensional frequency histogram. The cascade of such patterns over time produces so-called *patterns movies*, offering in a few seconds a visualisation of helicopter' parameters in operation over a long period of time, typically one year.

It is worth noting that the time dimension is lost on each time window w, making it possible to aggregate the studied parameters and to erase the local specificities. Each generated picture turns out to deliver a *time-agnostic* pattern, while the time dimension is still present in the "patterns movies". In other words, a patterns movie can be seen as a sequence of time-agnostic patterns, allowing to visualize how patterns slowly evolve over time.

A running example

In the sequel, we consider a running example to explain how a picture is built from two TS variables over one time window only, i.e. $w = \mathcal{T}$. The five steps described previously are exemplified on data depicted in Table 1.(a).

Step 1 (Discretization): We consider here a simple discretization, the rounding function $D(x) = \lfloor x \rceil$. For sake of readability, each pair of values in $\hat{T_1}$ and $\hat{T_2}$ has got a particular colour, as shown in Table 1.(b)

Step 2 (Counting): A new dimension is added to count how many times a given pair of values appears in $(\hat{T_1^w}, \hat{T_2^w})$, depicted in Table 1.(c).

Step 3 (Co-occurence matrix): A co-occurrence matrix is built. Rows refers to $\hat{T_1}$ values, i.e. $\langle 5, 6, 7, 8, 9 \rangle$, and columns to $\hat{T_2}$ values, i.e. $\langle 1, 2, 3, 4 \rangle$. A pair of values (u, v) refers to the counting associated to that pair in the time window

Table 1: Running example

(a)

T_1^w	5,3	4,7	5,5	5,8	6,1	7	8,2	8	8,3	8,6
T_2^w	1,9	1,5	3	3,4	4	3,1	3,6	3,9	4,2	0,5

$\Downarrow$ (b)

$\widehat{T_1^w}$	5	5	6	6	6	7	8	8	8	9
$\widehat{T_2^w}$	2	2	3	3	4	3	4	4	4	1

$\Downarrow$ (c)

$\hat{T_1^w}$	5	6	6	7	8	9
$\hat{T_2^w}$	2	3	4	3	4	1
Count	2	2	1	1	3	1

$\Downarrow$ (d)

$\hat{M}_w$		$\hat{T_2^w}$ values			
		1	2	3	4
$\hat{T_1^w}$ values	5	0	2	0	0
	6	0	0	2	1
	7	0	0	1	0
	8	0	0	0	3
	9	1	0	0	0

w (cf step 3). The co-occurrence matrix obtained is depicted in Table 1.(d). This representation plays the role of a two-dimensional histogram, where the time dimension is lost.

Step 4 (Picture generation): From the previous co-occurrence matrix of size $n \times m$, we sketch how pictures of size $n \times m$ can be generated. The main idea is that the larger the value of a matrix at (i, j), the darker the (i, j) pixel in the 2D-picture. In order to protect the observer from possible bias of reading and consequently of interpretation, we have adopted a normalization by distribution intervals to better reflect the real data density, i.e. the number of data per unit area [11].

A typical 2D-picture is given in Fig. 2, which corresponds to a pattern on two variables: the oil pressure and the oil temperature observed in the main gear box. The discretization is produced by partitioning the values of each attribute in $K = 100$ equal length intervals.

Step 5 (Pattern movie generation): This step is simple as it consists to generate such pictures over different time windows in order to produce a movie. The time dimension is taken into account here at a coarser granularity, allowing to study the global trends of different parameters. An example is shown in

Figure 2: Co-occurrence matrix 100x100, all in-flight data from aircraft A, "Oil temperature" *vs* "Oil pressure"

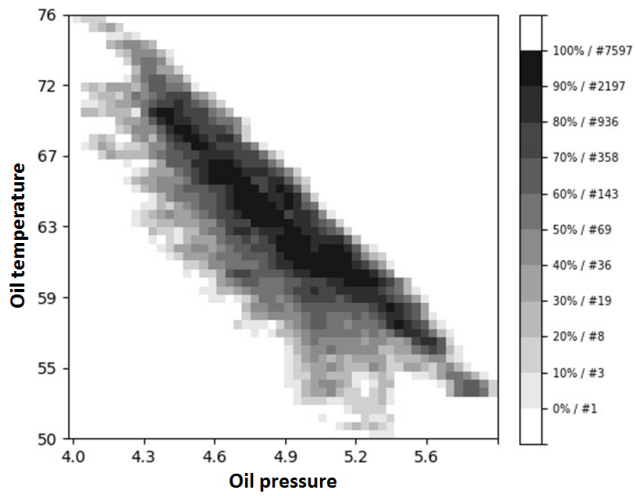


Figure 3.

In figure 3, we display some frames from a patterns movie built from data corresponding to only one aircraft. The studied TS variables are "Oil temperature" (x-axis) and "Oil pressure" (y-axis). The middle frame from the last row corresponds to a time window where an operating incident was reported.

Computational considerations

Generating patterns movies scales well over very large TS since each transformation has a complexity linear or quasi linear in the size of the input. Moreover parallelisation can be applied. Details are omitted.

3 Implementation and experimentations

The implementation has been done with `Python 3.6.7` with libraries `Pandas 0.24.1` for tabular data, `NumPy 1.16.2` for math operations, and `Matplotlib 3.0.3` for dataviz. Animated renderings use the FFmpeg encoder[3] for the generation of MPEG-4 video files[4].

Experiments were executed on an Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz with 16 Go RAM.

We studied several datasets, two of them are described below. The first one comes from a unique helicopter with over 45 days in operation, resulting in 76

[3]Bellard, F, FFmpeg, `ffmpeg.org`, 2019

[4]The Moving Picture Experts Group, M. MPEG-4 `mpeg.chiariglione.org/standards/mpeg-4/mpeg-4.htm`, 1998.

Figure 3: Snapshot of a patterns movie at different points in time

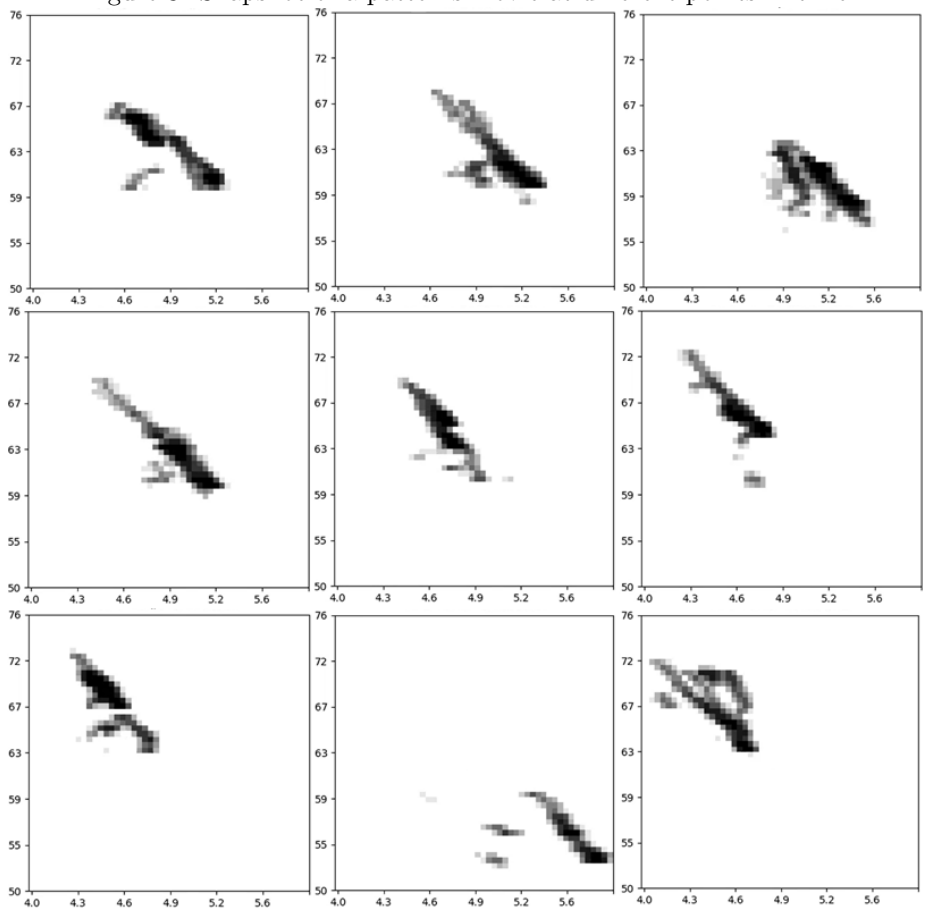
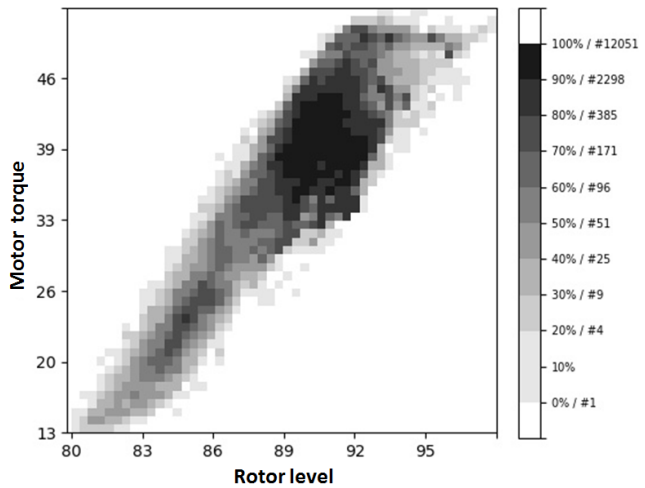


Figure 4: Co-occurrence matrix 100x100, all in-flight data from aircraft A, "Motor torque" *vs* "Rotor level"



flight hours and 550,000 records obtained at a frequency of 2 Hz. 24 parameters were recorded into attributes such as oil pressure, oil temperature or altitude. The second one was bigger with more than 118 million records, corresponding to 16,000 flight hours of 33 helicopters, recorded over a period of 20 months. On average we had 500 flight hours per aircraft.

The first dataset allowed an initial exploration by generating co-occurrence matrices between all possible pairs of attributes in a few minutes. For example, Figure 2 shows a relationship between the oil temperature and its pressure, with an equal-width dicretization of 100 bins in each dimension.

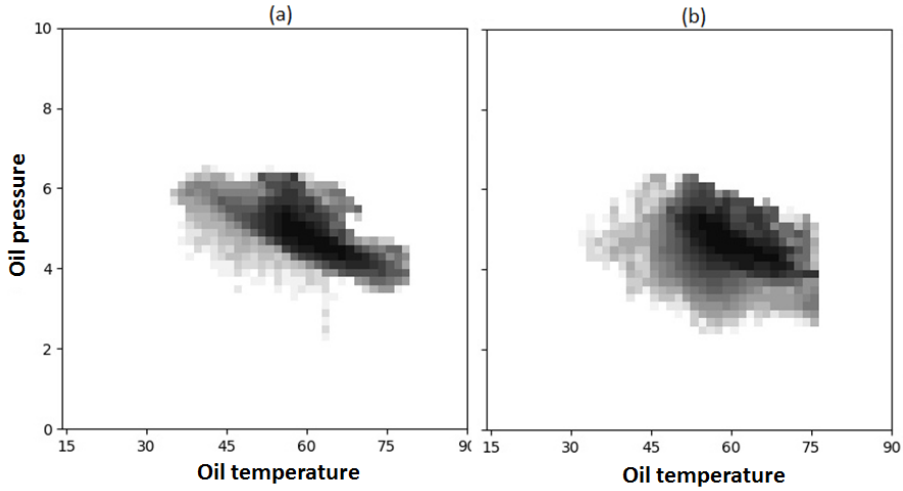
Similarly, Figure 4 shows a correlation between attributes playing a role in the mechanics of the helicopter. These first results were expected by business experts, and judged as promising to visualize TS data.

New visualizations of static and animated co-occurrence matrices were then regenerated on the new dataset. Two main observations can be drawn. First, on each device a global normality appears on the *kernels* of the representations, i.e. the region with the most frequent co-occurrences. We observe a great similarity of the most frequent data in Figure 5 (a) and (b) (the darkest part).

Second, data at the periphery of the kernel, ie the less frequent data, ranging from white to gray, do not follow the same distribution from one device to another, and represent a large region of the co-occurrence matrix. On the other side, the kernel part of this representation seems to be invariant.

Experts found this approach easy to use, giving them convenient and intuitive visualization of very large TS data. These results were useful in verifying hypotheses about the system behaviour, quite complementary of existing propositions [6], [8], [5].

Figure 5: Co-occurrence matrix 50x50, "Oil temperature" *vs* "Oil pressure", all in-flight data from aircraft B (a) and from aircraft C (b).



The prototype is intended to be used by SDR experts as a decision support system: It allows them to quickly comfort or invalidate their "implicit" hypothesis. Then, in case of comforted hypothesis, a more complex analysis and investigation are required in order to either precisely identify a root cause of an incident or tune a predictive maintenance indicator for the studied system.

4 Evolution in time of the centroid of co-occurence matrices

Splitting temporal data using time windows generates a multitude of successive ordered co-occurrence matrices. One of our goals is to better understand the evolution of these matrices with respect to the time dimension. The most intuitive way to do this is to compare the different matrices with each other. To do so, we look for a numerical measure between these matrices.

After exploring some classical distances adapted to matrices, like Jaccard or Manhattan, an euclidean distance based on matrix centroids has been chosen by domain experts for its simplicity of interpretation and visualization potential.

Let $C \in \mathbb{R}^{w \times h}$ be a matrix. The point $(\overline{x}, \overline{y})$ is the centroid of C if:

$$\overline{x} = \frac{\sum_{i=1}^{w} i \times \sum_{j=1}^{h} C(i,j)}{\sum_{i=1}^{w} \sum_{j=1}^{h} C(i,j)} \qquad \overline{y} = \frac{\sum_{j=1}^{h} j \times \sum_{i=1}^{w} C(i,j)}{\sum_{j=1}^{h} \sum_{i=1}^{w} C(i,j)}$$

For example, for the matrix:

$$\begin{bmatrix} 10 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 15 & 30 \end{bmatrix}$$

the corresponding centroid is:

$$\overline{x} = \frac{1 \times 10 + 2 \times 5 + 3 \times (15 + 30)}{10 + 5 + 15 + 30} = \frac{155}{60}$$

$$\overline{y} = \frac{1 \times 10 + 2 \times (5 + 15) + 3 \times 30)}{10 + 5 + 15 + 30} = \frac{140}{60}$$

Let $A, B \in \mathbb{R}^{w \times h}$ and $(\overline{x}_A, \overline{y}_A)$ and $(\overline{x}_B, \overline{y}_B)$ their corresponding centroids. We define the distance between A and B as the normalized euclidean distance between the corresponding centroids:

$$dist(A, B) = \frac{1}{\sqrt{w^2 + h^2}} \times \sqrt{(\overline{x}_A - \overline{x}_B)^2 + (\overline{y}_A - \overline{y}_B)^2}$$

The idea is to visualize the path followed by the centroids of co-occurrence matrices to get an idea of their movements over time. Centroids can be visualized statically by displaying all of them in order to have an idea of their os-called *transit zone*. They can also be visualized dynamically to show how they evolve over time.

More than providing information on the distances travelled by centroids, such visualizations also provide information on their direction and transit area. Transit zones can be used to represent normality and highlight centroids moving away from it.

The path followed by centroids before and after maintenance operations has been analyzed in order to detect trends and patterns with respect to the studied maintenance intervention (see figures 6 to 9).

Figure 6 shows the path of the centroids around a given n-th maintenance operation. The path of the centroids before the maintenance operation n (intervention on the dial) is drawn in red. The previous operation $n-1$ concerned an intervention on the gearbox. The operation $n+1$ concerns also an intervention on the dial, and the centroid path between n and $n+1$ is displayed in blue. Centroids are calculated on 50×50 oil temperature and pressure matrices, within a window size of 30 minutes. We observe a variation of the transit zone taken by the centroids over time.

The downside of the visualisation based on centroids is that it requires a lot of data to be relevant. If maintenance operations are too close in time, the small number of centroids tracing the path may not be sufficient to obtain a sufficiently precise idea of the transit zones they use.

Figure 7 and Figure 8 show the path of centroids around a maintenance operation on the gearbox lubrication of two different aircrafts. In red is the path before the operation, in blue, the path followed by the centroid after operation. We observe a similar displacement of the transit zone of the centroids towards a zone where the temperature is lower and the pressure higher. In this case

Figure 6: Centroids for co-occurrence matrix 50x50, all in-flight data from aircraft A, "Oil temperature" *vs* "Oil pressure", window size of 30 minutes

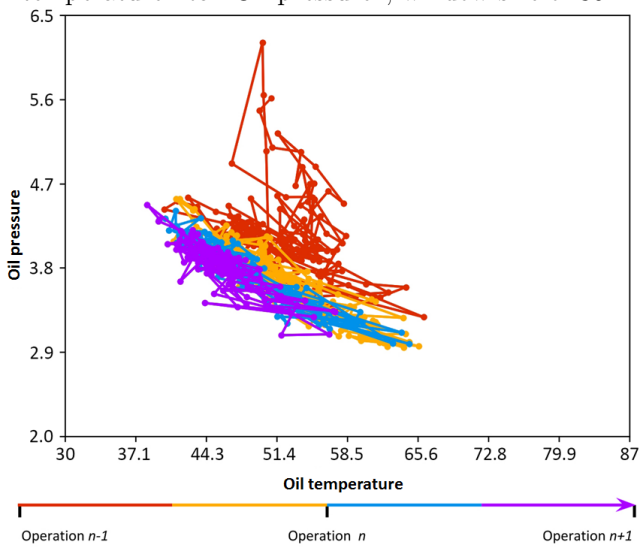


Figure 7: Centroids for co-occurrence matrix 50x50, all in-flight data from aircraft B, "Oil temperature" *vs* "Oil pressure", window size of 30 minutes

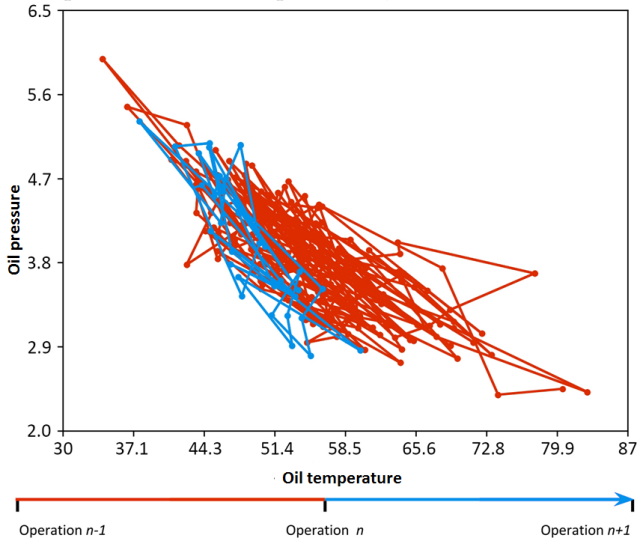


Figure 8: Centroids for co-occurrence matrix 50x50, all in-flight data from aircraft C, "Oil temperature" *vs* "Oil pressure", window size of 30 minutes

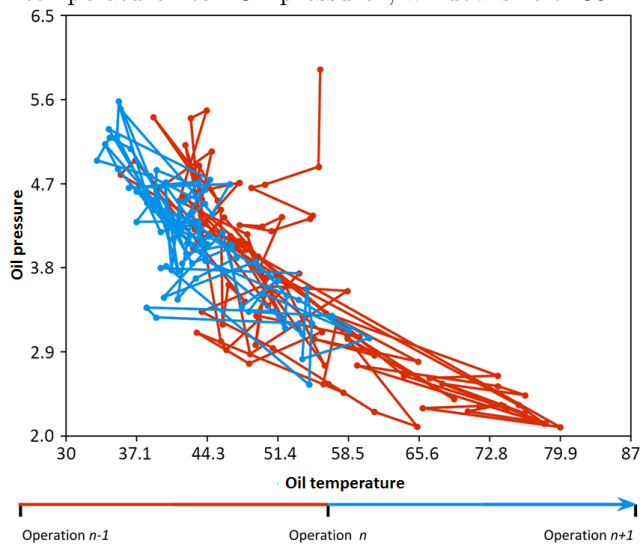
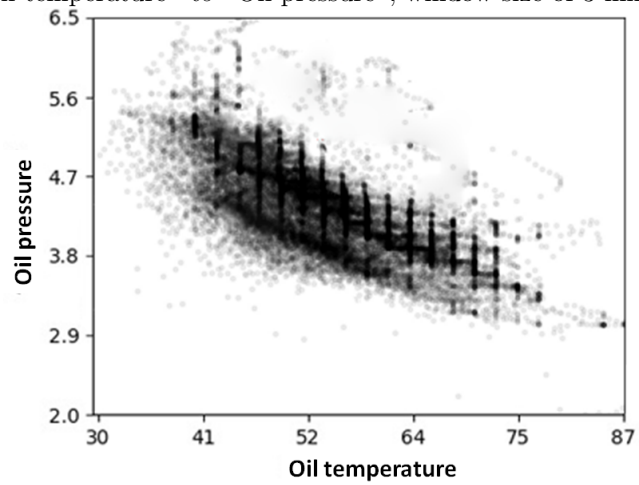


Figure 9: Centroids for co-occurrence matrix 50x50, all in-flight data from aircraft D, "Oil temperature" *vs* "Oil pressure", window size of 3 minutes



the path followed by the centroid is a good indicator for an human expert to validate the maintenance operation.

Some external factors are also influencing the centroid path. For example, the effect of an oil change operation may vary depending on the type of the used oil. Even if an operation was only to tighten a bolt on the device, the bolt could be tightened in many different ways. Nevertheless, we observe similar maintenance operations in our experiments.

The duration of the time windows has an impact on the visualization generated. Although the general trends for the transit zones used by centroids remain the same, the number of centroids and their local behaviors are affected by a change in the frequency of windowing. Too high a frequency would aggregate a lot of data and would result in a low number of centroids, perhaps too low to capture an evolution. Conversely, too fine a granularity would increase the number of centroids, which firstly lengthens the computation time, and can cause the appearance of artefacts. Figure 9 illustrates this phenomenon. It represents all the centroids of the 50x50 co-occurrence matrices, relating to the temperature of the oil and its pressure, obtained from a windowing with a frequency of 3 minutes. We observe the appearance of vertical line explained by the fact that the variability of the data on the oil pressure attribute becomes locally in time so small, that the horizontal position of the centroid is forced to fix on an integer value (the sensor precision is at 1 unit).

5 Conclusion

We introduced a technique for visualizing large TS data as patterns movies. On the basis of a division of the TS into time windows, co-occurrence matrices are built, allowing to display a representation of the underlying data distribution. The time dimension is lost locally at each matrix, but is kept globally throughout the windowing, in the produced pattern movie. Many experiments were conducted with TS data from Airbus Helicopters, from which we presented and discussed the main outcomes. In addition, we were able to aggregate further the visualization by focusing on centroids of co-occurence matrix only. Such an abstraction turned out to be very useful to study the normal behavior of the studied phenomenon with transit zone of centroids.

To sum up, this approach proposes to capture the evolution of trends observed over time windows in TS. Therefore, the main perspective is to be able to detect deviations with respect to the normal behavior of operating helicopters. Whenever these changes are detected, an alert could be raised to anticipate and better organize maintenance actions. The overall objective is to improve safety in order to avoid potential incidents, and allow customers to increase the availability of their helicopters. Experts found this approach very promising, intuitive, easy to use, allowing rapid testing of hypotheses on large collections of TS.

Many perspectives remain to be addressed: first, human perception of those pattern movies could be evaluated more thoroughly to define new visual quality

metrics [9]. Second, patterns movies could be used to anomaly detection, not by experts' eyes, but with automatic techniques on the co-occurrence matrices. Third, more research also deserves to be done to help domain experts to find appropriate tradeoffs to get meaningful pattern movies, for example to select the two parameters for the 2D visualization, to define an appropriate time window and also to discretize the data. Finally, co-occurence matrix can be extended to better handle the multivariate aspect, either by adding a set of dimensions for the x-axis and another set of dimensions for the y-axis, instead of a single dimension for both axis as we do; or by adding a set of dynamic indicators reflecting variations of other parameters during the last sliding windows.

Acknowledgements

Part of this work has been funded by the Datavalor initiative of the LIRIS laboratory.

References

[1] Boniol, P., Palpanas, T.: Series2graph: Graph-based subsequence anomaly detection for time series. Proc. VLDB Endow. **13**(11), 1821–1834 (2020)

[2] Dang, X., Shah, S.Y., Zerfos, P.: seq2graph: Discovering dynamic non-linear dependencies from multivariate time series. In: 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, December 9-12, 2019. pp. 1774–1783. IEEE (2019)

[3] Esling, P., Agón, C.: Time-series data mining. ACM Comput. Surv. **45**(1), 12:1–12:34 (2012)

[4] Haralick, R.M., Shanmugam, K.S., Dinstein, I.: Textural features for image classification. IEEE Trans. Syst. Man Cybern. **3**(6), 610–621 (1973)

[5] Keim, D.A., Mansmann, F., Schneidewind, J., Thomas, J., Ziegler, H.: Visual analytics: Scope and challenges. In: Simoff, S.J., Böhlen, M.H., Mazeika, A. (eds.) Visual Data Mining - Theory, Techniques and Tools for Visual Analytics, Lecture Notes in Computer Science, vol. 4404, pp. 76–90. Springer (2008)

[6] Kumar, N., Lolla, V.N., Keogh, E.J., Lonardi, S., Ratanamahatana, C.A.: Time-series bitmaps: a practical visualization tool for working with large time series databases. In: Kargupta, H., Srivastava, J., Kamath, C., Goodman, A. (eds.) Proceedings of the 2005 SIAM International Conference on Data Mining, SDM 2005, Newport Beach, CA, USA, April 21-23, 2005. pp. 531–535. SIAM (2005)

[7] Mechouche, A., Daouayry, N., Cameini, V.: Helicopter big data processing and preditive analytics: feedback and perspectives. In: European Rotorcraft Forum, Warsaw, Poland, Sept. 2019. p. 6 pages (2019)

[8] Peng, R.: A method for visualizing multivariate time series data. Journal of Statistical Software, Code Snippets **25**(1), 1–17 (2008)

[9] Tatu, A., Bak, P., Bertini, E., Keim, D.A., Schneidewind, J.: Visual quality metrics and human perception: an initial study on 2d projections of large multidimensional data. In: Santucci, G. (ed.) Proceedings of the International Conference on Advanced Visual Interfaces, AVI 2010, Roma, Italy, May 26-28, 2010. pp. 49–56. ACM Press (2010)

[10] Torkamani, S., Lohweg, V.: Survey on time series motif discovery. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. **7**(2) (2017)

[11] Tufte, E.R.: The visual display of quantitative information. Graphics Press (1992)