POLITECNICO DI TORINO
Repository ISTITUZIONALE

GA-Adaptive Template Matching for Offline Shape Motion Tracking Based on Edge Detection: IAS Estimation from the SURVISHNO 2019 Challenge Video for Machine

Original
GA-Adaptive Template Matching for Offline Shape Motion Tracking Based on Edge Detection: IAS Estimation from the SURVISHNO 2019 Challenge Video for Machine Diagnostics Purposes / Daga, A.P., Garibaldi, L.. - In: ALGORITHMS. - ISSN 1999-4893. - 13:2(2020). [10.3390/a13020033]

Availability:
This version is available at: 11583/2786432 since: 2020-01-29T15:17:07Z

Publisher:
MDPI

Published
DOI:10.3390/a13020033

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

20 November 2026

algorithms

MDPI

Article

GA-Adaptive Template Matching for Offline Shape Motion Tracking Based on Edge Detection: IAS Estimation from the SURVISHNO 2019 Challenge Video for Machine Diagnostics Purposes

Alessandro Paolo Daga * and Luigi Garibaldi

Dipartimento di Ingegneria Meccanica e Aerospaziale—DIMEAS, Politecnico di Torino, Corso Duca degli Abruzzi, 24, I-10129 Torino, Italy; luigi.garibaldi@polito.it

* Correspondence: alessandro.daga@polito.it

Received: 11 December 2019; Accepted: 25 January 2020; Published: 29 January 2020

Abstract: The estimation of the Instantaneous Angular Speed (IAS) has in recent years attracted a growing interest in the diagnostics of rotating machines. Measurement of the IAS can be used as a source of information of the machine condition per se, or for performing angular resampling through Computed Order Tracking, a practice which is essential to highlight the machine spectral signature in case of non-stationary operational conditions. In these regards, the SURVISHNO 2019 international conference held at INSA Lyon on 8–10 July 2019 proposed a challenge about the estimation of the instantaneous non-stationary speed of a fan from a video taken by a smartphone, a pocket, low-cost device which can nowadays be found in everyone's pocket. This work originated by the author to produce an offline motion-tracking of the fan (actually, of the head of its locking-screw) and obtaining then a reliable estimate of the IAS. The here proposed algorithm is an update of the established Template Matching (TM) technique (i.e., in the Signal Processing community, a two-dimensional matched filter), which is here integrated into a Genetic Algorithm (GA) search. Using a template reconstructed from a simplified parametric mathematical model of the features of interest (i.e., the known geometry of the edges of the screw head), the GA can be used to adapt the template to match the search image, leading to a hybridization of template-based and feature-based approaches which allows to overcome the well-known issues of the traditional TM related to scaling and rotations of the search image with respect to the template. Furthermore, it is able to resolve the position of the center of the screw head at a resolution that goes beyond the limit of the pixel grid. By repeating the analysis frame after frame and focusing on the angular position of the screw head over time, the proposed algorithm can be used as an effective offline video-tachometer able to estimate the IAS from the video, avoiding the need for expensive high-resolution encoders or tachometers.

Keywords: machine vision; machine diagnostics; instantaneous angular speed; SURVISHNO 2019 challenge; video tachometer; motion tracking; edge detection; parametric template modeling; adaptive template matching; genetic algorithm

1. Introduction

Rotating machinery are fundamental components of mechanical systems for most of the industrial applications, as mechanical power is commonly obtained in the form of torque at a rotating speed. Electrical motors, internal combustion engines and motors in general, in fact, convert some

sort of source energy into mechanical energy which is transferred to the final user through a mechanical transmission (e.g., a gearbox) which provides speed and torque conversion.

The speed is then a piece of essential system information, usually kept under control to accomplish a determined work. As a general consideration, feedback controllers can easily maintain their process variable (i.e., the speed in this case) close to its setpoint, but noise and disturbances may affect the controlled system instantaneous output causing the variable to depart from and oscillate around its desired value. The appearance of damages in the system has then repercussions on the instantaneous speed, which can be used as a source of information of the machine health condition perse.

The information about the instantaneous speed is fundamental for machine diagnostics also in other ways. In the field of Vibration Monitoring (i.e., a particularly successful kind of condition monitoring based on vibration records) for example, it is very common to use measured or estimated information about the Instantaneous Angular Speed (IAS) to perform the so-called Order Tracking [1]. In fact, most of rotating machines are affected by phenomena which are locked to particular angular positions (e.g., the intake of a 4 strokes diesel engine lasts from 0 to π radians of the main shaft and is periodic of 4π; a gear featuring a wheel with M teeth whose N^{th} tooth is damaged features an anomalous meshing pattern at angle $\frac{2\pi N}{M}$ of the wheel's shaft with a periodicity of 2π). Furthermore, in the case of non-stationary acquisitions, the spectral signature of the machine can only be found after synchronization of the Fourier analysis on the shaft rotation (i.e., the spectrum is not given as a function of the frequency, but as a function of the orders of a reference shaft). Unfortunately, commercial Data Acquisition Systems (DAQ) are not efficient in sampling at constant angular increments of the reference shaft, so that resampling algorithms are used, based on the additional information of the angular position of the reference shaft over time [2–5].

The growing interest of the signal processing community in the IAS information is demonstrated by the increasing number of publications involving the estimation of the instantaneous frequency as well as the creation of a special issue on IAS processing in the Journal of Mechanical Systems & Signal Processing [6] and of a conference dedicated solely to condition monitoring in non-stationary operations [7]. In addition, at the international conferences Surveillance 8 (in 2015) [8] and SURVISHNO (in 2019), contests about IAS estimation were organized. The IAS is in effect spreading in many fields of diagnostics of rotating machinery, especially for motors and gearboxes.

Particular applications for motors diagnostics involve electrical motors [9–11], internal combustion engines [12–16], and hydraulic engines too [17].

Gearbox diagnostics was also covered from a general point of view [18–21] as well as considering particular applications (e.g., windmills [22]) or focusing on relevant components such as the bearings [23,24].

Machining processes can also be subject to diagnostic analysis through the IAS estimation. The milling cutting force, for example, is proved to be reflected by the IAS [25,26].

1.1. Instantaneous Angular Speed (IAS) Review

The IAS information is, in simple terms, a measure of the rotation speed of a rotating component of a machine defined at an angular resolution corresponding to at least one value per revolution [26]. From a physical point of view then, the IAS is defined starting from the angular position $\alpha(t)$ of a shaft as:

$$\omega(t) = \dot{\alpha}(t) = \frac{d\alpha}{dt} \tag{1}$$

A measure of this quantity can be obtained using analog sensors such as the tachometer-dynamo which translates the rotational speed into an electrical signal, or with analog angular position sensors such as the resolver. Nevertheless, such devices can nowadays be considered obsolete and are often substituted by more reliable digital sensors, whose signals can be processed to produce an estimate:

$$\widehat{\omega}(t) \cong \frac{\Delta\alpha}{\Delta t}\ ;\ \widehat{\omega} \to \omega\ for\ \Delta t \to 0 \tag{2}$$

In particular, two strategies can be put in place. One involves the measurement of the angle $\Delta\alpha$ swept in a constant interval of time Δt by the shaft. Nevertheless, the angular measurements can be

affected by larger errors than the time measurements' ones. A second more widespread and accurate method consists of measuring the time elapsed between successive pulses (Elapsed Time $ET = \Delta t$) corresponding to a known swept angle $\Delta\alpha$ which is a characteristic of the sensor (e.g., an encoder). An analysis of the resolution and the speed estimation error can be found in [27].

In any case, all these methods involve the use of an additional sensor (generally referred to as a generic "tachometer" but not to be confounded with the tachometer-dynamo sensor). The interest of the scientific diagnostics community then has recently moved to IAS estimation from the easily available accelerometric signals, fundamental for vibration monitoring, going "encoder-less" or "tacholess". In this regard, several different approaches are possible [7].

In particular, the simplest idea is to track a shaft-speed related harmonic (possibly showing a good Signal to Noise Ratio-SNR) from its corresponding peak in a time-frequency representation of the signal (i.e., a Spectrogram, computed for example via a Short Time Fourier Transform). This can be made more accurate by averaging the tracks from multiple harmonic orders, which can also be automatically selected by the algorithm (e.g., Multi-Order Probabilistic Approach-MOPA, Ceptrsum-based MOPA, or ViBES).

A second approach to the IAS estimation is the demodulation of a shaft-speed related harmonic exhibiting good SNR. This follows the idea that vibration signals can be modulated by the revolution of the shaft so that phase demodulation can recover the shaft speed from the band-pass isolated harmonic of interest.

Demodulation can also be performed using the Teager-Kaiser Energy Operator.

Finally, the two main procedures of tracking and demodulation can be exploited together as implemented in the Vold-Kalman filtering.

At any rate, both the measurement and estimation of the IAS have some limitations. In the first case, the use of high-resolution encoders allows to get reliable and accurate estimates of the IAS, but the angular sensors can be very expensive and need to be mounted on a shaft added for the purpose. On the other hand, the second case exploits the cheap and reliable accelerometers which can be added on a machine without special design updates. The estimated IAS, however, is less accurate and less reliable, and needs computational time.

In this article then, the measurement of the IAS is tackled at the scope of testing an offline, low-cost video-tachometer approach, as suggested by the SURVISHNO 2019 challenge (see Supplementary Material). The question was "how far is it possible to carry out relevant analysis from a video or a rotating fan acquired by a smartphone which can reach a maximum rate of 30 frames per second?".

In order to answer the question, the field of computer vision was explored so as to point out the main approaches to shape detection and image recognition at the scope of extracting the angular position $\alpha(t)$ of the fan from the video and then obtain an estimate of the IAS.

1.2. Brief Literature Review of Computer Vision

Computer (or Machine) Vision is a scientific discipline that deals with Machine Learning applied to digital images and videos coming from cameras as well as any other visual representation derived from various sensors such as ultrasonic cameras, range sensors, radars, tomography devices, etc. The objective is transforming visual images into descriptions of the world by extracting data and features, producing then information which is fundamental in decisional processes. Applications of Machine Vision to industrial processes include automatic inspection (e.g., to detect manufacturing defects), surveillance and security (e.g., detection of events, face recognition, etc.), motion control, navigation and human-machine interaction (e.g., robots, autonomous vehicles, etc.), modeling objects or environments, improving human vision (e.g., medical image analysis and detection), or organizing information (e.g., image classification databases, etc.).

Anyway, according to the need of the particular application, this article is interested only in a small sub-domain of Machine Vision algorithms, namely object recognition (i.e., finding and identifying objects in an image or video), video tracking (i.e., locating a moving object over time using a camera), and motion estimation (i.e., determining motion of a body from two following 2D frames).

In order to obtain the angular position of the fan $\alpha(t)$, in fact, all the three domains are needed. First, the subject must be identified in a frame. Then, its position and orientation should be compared to the subject in previous or in a reference frame. This way the motion can be tracked from frame to frame. In general, two main categories of algorithms can be found in the literature [28–30]. Direct methods are based on the pixel information, while indirect methods make use of features such as corners or particular points of the subject. Feature-based methods minimize an error measure based on distances in the feature space, while direct methods minimize an error measure based on direct image information collected from all pixels in the image (i.e., typically the pixel brightness, which is usually computed from the RGB color image at a pre-processing stage).

Complete reviews of approaches to image processing and face recognition (i.e., a common application of object recognition) can be found in [31–33] from which it is clear that Neural Networks strongly entered the game of machine vision (in particular for feature-based methods). However, all the reviews also agree in pointing out a common foundation of the processing: Template Matching (TM).

Template Matching [34] is a technique in digital image processing for finding small parts of an image which match a template image (i.e., an image designed to serve as a model). TM is often considered a very basic and limited approach to computer vision, but it is actually involved in many old and new techniques. In [35] for example, TM is taken into account only for reference-comparison inspection of mass-produced integrated circuits precisely aligned on a conveyor (i.e., equal objects, with the same location, scale, and orientation in the image). In fact, the limitations of traditional TM are well known [34–36] and can be summarized as: a) noise, illumination changes, and occlusions, b) background changes and clutter, c) rigid and non-rigid transformations and scale changes (i.e., images are a projection of a 3D scene onto a 2D plane), d) high computational cost.

Point a) is commonly tackled first at a pre-processing stage, when the RGB color image is translated into a pixel-wise single channel brightness information (i.e., the luminance) and edge detection is performed, and later by focusing on the stability and robustness of the selected similarity measure, which affects also point c). Examples of standard similarity measures are the pixel-wise sum of differences of the search image and the template, the sum of products (or cross-correlation) among them, the Best-Buddies-Similarity based on Nearest-Neighbor matches of features, the Deformable Diversity Similarity [36]. Limitations in b) and d) are often faced with the simple trick of using masks to remove non-interesting areas (i.e., providing a search window), or otherwise, reducing the number of sampling points by decreasing the resolution. Point c) was sometimes dealt with the implementation of multiple templates with different scales and rotations (e.g., eigenspaces) or with Deformable Part Models (DPM) or Deformable Template Matching [37]. This paper anyway focuses on the exploitation of the Genetic Algorithm for dealing with rigid transformations and scale variations in the template.

1.3. GA and Template Matching: A Review

Evolutionary Algorithms are recently enjoying new success in the scientific community for generating good solutions to optimization and search problems without relying on assumptions about the underlying fitness landscape (i.e., they perform derivative-free optimization).

A Genetic Algorithm (GA) is a heuristic algorithm inspired by Charles Darwin's theory of natural evolution via natural selection, where the best individuals are more prone to reproduction and have better offspring.

When focusing on TM, GA can be found in several applications and in different fields, from manufacturing (i.e., Integrated Circuits quality inspections) to security and surveillance (i.e., animals recognition or face recognition) up to medicine (i.e., nodulus recognition in Computed Tomography CT scans) as found in [38–42]. In particular, in [38] GA is used to speed up the TM by shrinking the image to be processed. More refined employment of GA can be found in [39], where face recognition is performed by TM using a T-shaped template isolating eyes, nose and mouth, resized by GA to find a better match when the size of search image and template is different. In [40] further improvement is proposed, as a Deformable Template generated as a Point Distribution Model (PDM) is adapted by

GA to measure characteristic landmark points (i.e., vertices, nodes, markers, etc.) on cattle images for morphological assessment of bovine livestock. The idea of generating a template through a model optimized by GA was applied also in [41] and in [42] for automatic detection of lung nodules in chest spiral CT scans. Nevertheless, all three applications show some weaknesses. In [40], because of the scope of the analysis, the deformable template is GA adapted to locate landmark points on the cattle PDM profile, but this does not allow to extract clear and unique information about scale, orientation, or center of the template shape, which remains a parametric function (i.e., the template is not a digital image). In [41], a two-step GA is proposed for real-time shape tracking. The first step optimizes the template which is not directly generated by a mathematical model but comes from a mathematical mask (characterized by 3 parameters) on a digital image template, while the second optimizes orientation and center position the template (but does not accounts for the scale). Nevertheless, the higher computational efficiency of using a two-steps GA rather than a single step GA is not justified. Finally, in [42], the template is actually halfway between the parametric function and the digital image: a simple parametric function is used in this case to generate multiple images which will be used as multiple templates (different scale and rotation). Nevertheless, GA is set up to perform a discrete rather than a continuous search of the best matching template in terms of rotation, scale, and center position. This limits the potential of GA-TM of getting sub-pixel accuracy [30].

As a result of these considerations, a novel GA-adaptive TM technique is proposed in this work. In particular, GA is integrated into the TM so as to reconstruct a digital template from a simplified parametric geometrical model (which acts as a mask) whose parameters can be continuously optimized in terms of scale, rotation, and center position so as to maximize a similarity measure and to find the best match. This not only overcomes the well-known issues of the traditional TM related to deformations in the search image with respect to the template but enables to resolve the position of the center at a resolution that goes beyond the limit of the pixel grid, allowing an effective shape tracking which is used in this paper for implementing an offline video-tachometer able to estimate the IAS of the SURIVSHNO 2019 fan from the video, avoiding the need of expensive high-resolution encoders or tachometers.

2. Materials and Methods

This work is meant to propose an inexpensive but effective video-tachometer using a 30 fps (i.e., frames per second) video from a mobile phone. The target is the IAS estimation of the SURIVSHNO 2019 fan. The raw dataset is then composed by a bunch of sequential digital color pictures, each of them corresponding to a matrix of pixels (i.e., "picture element": the smallest addressable element of the digital image) updated during a full scan of the camera image sensor. The dataset is described in Section 2.1. In Section 2.2 the principle of TM is introduced, while in Section 2.3 the GA is integrated into TM. Finally, the overall methodology for estimating the IAS is reported in Section 2.4.

2.1. Data Description: the SURVISHNO 2019 Challenge Video and Its Critical Issues

As already introduced, Computer Vision deals with the understanding and interpreting of visual representations such as digital images and videos, as well as other representations which will not be considered in this paper. A video is the electronic medium for recording and displaying a moving visual media, namely a chronographic sequence of photographic shots which forms a representation of the visual world and can capture motion. The representation of the visual characteristics of an object is converted by image sensors into digital signals that can be processed by a computer and made output through a screen as a visible-light image. The 2D digital image is spatially discretized in a number of addressable elements (i.e., the pixels-px) organized in rows and columns to cover the entire image space. The standard full-HD High-Definition Television (HDTV) system uses a resolution of 1920 × 1080 px with 16:9 aspect ratio, so that each of the 2,073,600 pixels stores three-channels color information. Trichromacy, in fact, mimics the animal vision which uses three different types of cone cells in the eye to perceive not only light intensity but also its spectral composition (i.e., color). A very common set of primary colors is that defined by the RGB color model, an additive model in which red, green, and blue light are added together to reproduce a broad range

of colors. In particular, graphics file formats usually store RGB pictures as 24-bit images (i.e., RGB24 format), where RGB components are 8 bits each, so that each color intensity can be rendered at 256 levels (normalized to unity between 0 and 1 or, more commonly, with integers between 0 and 255), leading to a potential of 16,777,216 (about 16 million) colors.

Therefore, in digital imaging systems (e.g., digital cameras, mobile phones, etc.) the acquisition corresponds to an interrogation of each pixel photo-sensor so that the full image is recorded; to produce a video this image recording is repeated in time at a given sampling frequency (commonly 30 fps). Nevertheless, the pixel sensors can be either interrogated simultaneously (i.e., global shutter) or, more frequently in mobile phones, one after the other in a predetermined sequence (i.e., rolling shutter).

In the particular case, the SURIVSHNO 2019 video is a sequence of 1298 RGB24 images acquired at 30fps for a duration of 43.3 s. All the frames are recorded at a full-HD resolution of 1920 × 1080 px with 16:9 aspect ratio. The RGB color image depicts a front view of a fan composed by 10 blades, coupled to a spindle by a hexagonal shaft and an 8.8 screw. The first frame of the video is reported in Figure 1.

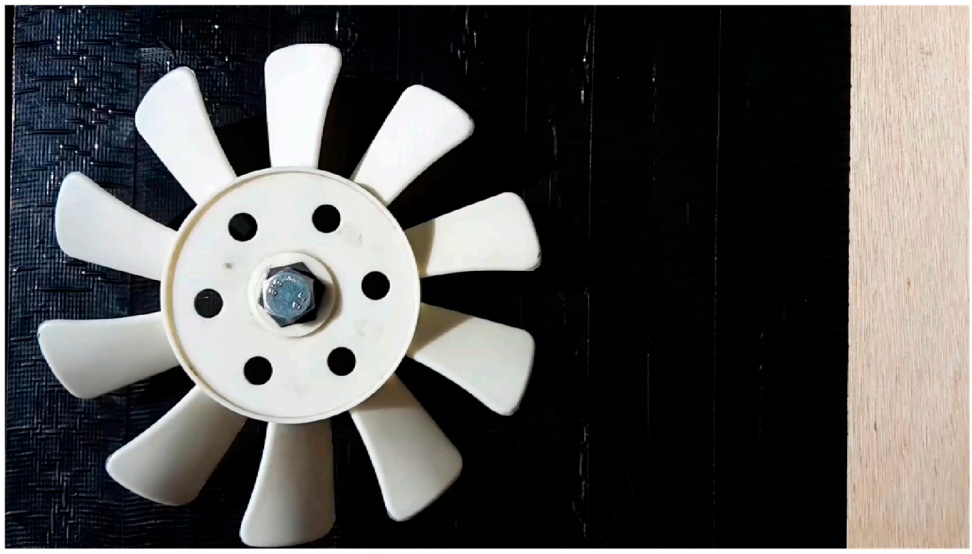


Figure 1. The first frame of the video, digitalized as a 1080 × 1920 × 3 matrix: full-HD resolution of 1920 × 1080 (16:9 aspect ratio), where each pixel color is represented as a triplet of intensities for the three-color components namely red, green, and blue (RGB additive trichromacy color model).

The IAS of the fan is unknown, nevertheless, it can be recognized as strongly non-stationary by watching the video. Some typical issues of non-stationary cases, in fact, arise. In particular, spatial aliasing due to the rolling shutter effect of the camera can be easily noticed. As the fan is spinning counterclockwise at an increasing rotational speed, the blades on the left side appear to get thicker while the blades on the right side appear to become thinner as the video progresses in time. This is visualized in Figure 2, in contrast to Figure 1.

Aliasing is a typical issue of digital signals. A digitally reconstructed image, in fact, will differ from the original image (i.e., analog) because of the spatial discretization (i.e., the sampling) so that visible patterns or deformations can compromise the quality of the reconstruction.

Temporal aliasing, determined by the sampling frequency or, in case of videos, by the frame rate of the camera, is a major concern of Digital Signal Processing. In videos, because of the limited frame rate (N.B., limited with respect to the rotating speed of the object), a rotating object like a fan or a wheel looks like turning in reverse or too slowly. A similar effect is probably experienced by any human beings in the form of an optical illusion called "wagon-wheel effect" which may occur even under truly continuous illumination because of the human visual perception.

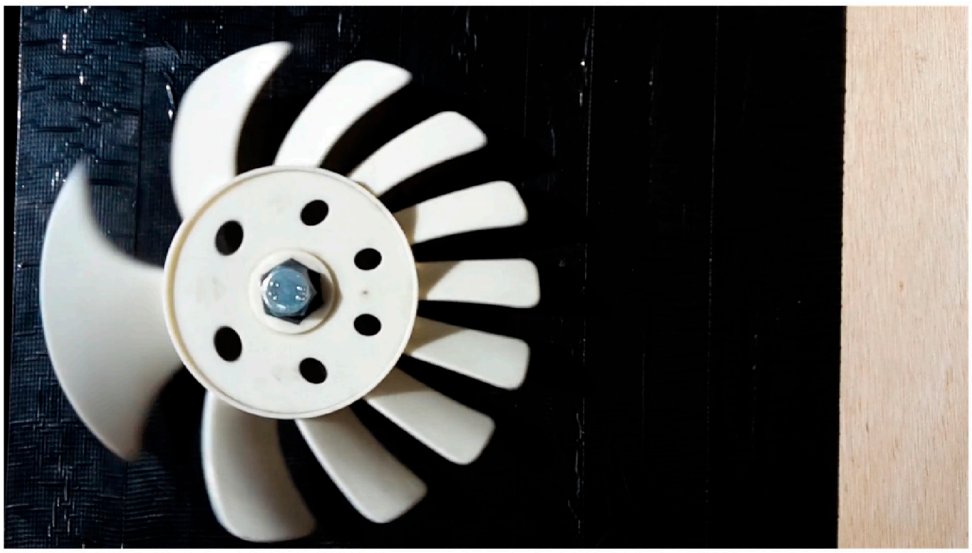


Figure 2. The thousandth frame of the video. The rolling shutter effect is highlighted as a deformation of the blades.

Sampling at 30 fps, in accordance with the Nyquist sampling theorem, allows to correctly picture phenomena which are bandlimited to half the sampling rate (i.e., 15 Hz, the Nyquist frequency) without aliasing. In the SURIVSHNO 2019 video, the fan starts from a standstill and accelerates up to values lower than the Nyquist frequency, so that temporal aliasing does not occur. Nevertheless, by looking at the video, the optical illusion of a reverting direction of rotation occurs anyway as the brain cannot recognize the 10 equal blades of the fan, so the exceeding of 1,5 Hz causes a reversal of the perceived direction of rotation.

A final issue is related to the autofocus of the camera. When taking photos, in fact, a convex lens is used in the camera to focus incoming light onto a photo-sensor array (e.g., a Complementary metal-oxide-semiconductor-CMOS-photo-sensor). In order to ensure crisp and clear images, the optical system commonly uses a control system and a motor to optimize the distance between the lens and the sensor. This can obviously lead to distortions of the image during the video.

To summarize, three main issues should be tackled:

- Spatial Aliasing related to the rolling shutter effect,
- Temporal Aliasing due to the 30-fps sampling rate given the 10 equal blades of the fan,
- Additional autofocus distortions.

Nevertheless, a workaround can be found to simplify things. First, the spatial aliasing occurs when the object moves faster than a limit speed dictated by the rolling shutter clock. Being the fan is rotating around its center, the higher the distance from the center, the higher the tangential speed, so that, focusing on a part of the image very near to the center (i.e., the fan-locking screw head), the spatial aliasing effect is minimal.

Second, the temporal aliasing, in this case, is more a visualization issue rather than a real problem for the analysis. The fan speed, in fact, is always lower than 10 Hz, so that if the attention is brought to a feature that occurs just once per revolution (i.e., the 8.8 logo on the fan-locking screw head) rather than the blades (which are 10 and not distinguishable), no temporal aliasing occurs.

Finally, the autofocus distortions can be accounted for, together with other perspective distortions by implementing the adaptive TM introduced in the following sections. In the SURVISHNO 2019 challenge video acquisition, in fact, the camera was almost but not perfectly aligned to the fan axis, so that, during the revolution, the center of rotation moves in a small region, while the image undergoes slight deformations.

2.2. Matched Filters and Template Matching

The problem of finding parts of a search image which match a template image is just a bi-dimensional extension of the common unidimensional Signal Processing (SP) problem of detecting the presence of a template signal in a search signal, typically a noise-affected measurement. This problem was first solved in the mid-40s by North, Vleck, and Middleton [43–46] as a response to the immediate need to improve radar performance during World War II [43]. In the original framework, the issue with radar is to highlight the presence of an echo (i.e., a known template) exhibiting little power and obscured by noise in a received signal (i.e., the search signal). Assuming a Gaussian white noise (i.e., with a flat power spectrum), the noise contributes with equal undesired power at all frequencies, while the signal, on the contrary, shows a bandlimited spectral content. Considering a transmitted pulse (i.e., a rectangular function), its spectrum is described by a sinc pulse (i.e., a sinc function) which is theoretically defined over the whole frequency axis but has practically most of the power bound to low frequencies. A matched filter is then a linear time invariant filter that maximizes the signal-to-noise power ratio highlighting then the presence of the template in the search signal. Intuitively, it is then a filter that emphasizes the frequency where the template power is contained (i.e., the low frequencies) while attenuating those where the only noise is present. If the template is known then, it is sufficient to use the template spectrum for designing the best filter frequency response.

The filter impulse response to be convolved with the search signal (i.e., for discrete signals, $s[n]$, where n is the sample index related to time by the sampling period) is then just the time-reversed version of the template $\hat{t}[n]$ of finite length N (N.B., more in general, for complex search signals, the conjugated time-reversed). The filter is then said to be "matched" to the template. See Equations (3) and (4) for the discrete Matched Filter impulse response definition ($h[n]$).

In the Matched Filter output a peak occurs (i.e., the amplitude goes "considerably" greater than the rest of the output signal $y[n]$ in the time domain) when the template signal is detected. By playing a bit with the notation, the convolution of the search signal with the time-reversed version of the template ($s[n] * h[n]$ in Equation (5)) is equivalent to the cross-correlation of the template (as it is) with the search signal ($r_{ts}[n]$ in Equation (6)). In the same way, if the cross-correlation shows a peak for a given delay (or lag) k, then the template is detected.

$$\begin{cases} t[n] = \hat{t}[n] & \begin{matrix} s[k] & -K < k < K \\ & n = 0, \dots, N-1 < K \end{matrix} \\ t[n] = 0 & -K < n < 0 \ \vee \ N-1 < n < K \end{cases} \tag{3}$$

$$h[n] = t[-n] \tag{4}$$

$$y[n] = s[n] * h[n] = \sum_{k=-K}^{K} h[n-k]s[k] \tag{5}$$

$$y[n] = \sum_{k=-K}^{K} t[k-n]s[k] = \sum_{k=-K}^{K} t[k]s[k+n] = r_{ts}[n] \tag{6}$$

The same consideration holds also when the problem is extended to a discrete 2D search signal such as a monochromic image where the light intensity $s[n, m]$ is a function of the spatial coordinates n and m over the pixel grid (i.e., a matrix). This problem is referred to as TM. Given a known template $\hat{t}[n, m]$ of size $N \times M$, it is possible to perform cross-correlation by simply moving the center of the template over each pixel of the search image pixel-grid of size $J \times K$ and calculate the sum of the pixel-wise products of s and $\hat{t}$ over the area spanned by the template $\hat{t}$. In a more rigorous formulation:

1. The template $\hat{t}$ is placed at $(n_0 + N/2, m_0 + M/2)$ in a matrix t of the same size $[n, m]$ of the search matrix $s[n, m]$ (Equation (7)),
2. The entrywise product (also known as Hadamard or Schur product, here "$\circ$") is performed finding the matrix $st[n, m]$ (Equation (8))
3. The correlation r_{ts} at $(n_0 + N/2, m_0 + M/2)$ is obtained by summing all the components in $st[n, m]$ (Equation (9))
4. By letting n_0 and m_0 vary in the range $n_0 = 1, \dots, J - N + 1 \vee m_0 = 1, \dots, J - M + 1$, the whole cross-correlation matrix is computed.

$$\begin{cases} t[n,m] = \hat{t}[n,m] & \begin{matrix} s[j,k] \quad 1 < j < J \ \vee 1 < k < K \\ n = n_0, \dots, n_0 + N \ < J \ ; \ m = m_0, \dots, m_0 + M < K \end{matrix} \\ t[n,m] = 0 & \begin{matrix} 1 < n < n_0 \vee n_0 + N < n < J \ ; \ 1 < m < m_0 \vee m_0 + M < m < K \\ n_0 = 1, \dots, J - N + 1 \ \vee m_0 = 1, \dots, J - M + 1 \end{matrix} \end{cases} \quad (7)$$

$$st[n,m] = s[n,m] \circ t[n,m] \quad (8)$$

$$r_{ts}[n_0 + N/2, m_0 + M/2] = \sum_{j=1}^{J} \sum_{k=1}^{K} st[j,k] \quad (9)$$

If the template is present in the search image, the cross-correlation features a maximum. The template detection becomes then a search for the maximum. This implies that the cross-correlation can be used as an effective similarity measure.

Nevertheless, this basic approach to TM is effective only when the template is a crop from an acquired reference image, and the search image is acquired under the same conditions (i.e., illumination, scale, orientation, fixed background, etc.). That is why in [35] TM is considered only for quality inspection of precisely aligned integrated circuits.

The limitations of traditional TM, as already introduced in Section 1, are well known [34–36] and are here summarized:

a) noise, illumination changes, and occlusions in the search image,
b) background changes and clutter,
c) rigid and non-rigid transformations, rotations, and scale changes (i.e., images are a projection of a 3D scene onto a 2D plane),
d) high computational cost.

Nevertheless, in this particular application (i.e., the SURVISHNO 2019 fan), as introduced in the previous sub-section, it was proposed to solve the issue of aliasing by focusing on the fan-locking screw head region. This reduction of the search space performed by cropping the image around the center of rotation is beneficial also according to points b) and d), as the background is effectively removed, while the computational burden is lightened.

The overall pre-processing is described in the next subsection, which treats about the edge detection performed on the basis of the monochromic image obtained from the brightness of the original RGB24 image. This helps in relieving the issues in point a).

Finally, point c) is addressed by exploiting the GA for dealing with rigid transformations and scale variations of the template to obtain a better match.

2.3. Image Preprocessing

In order to maximize the performance of the TM, a preprocessing of the image is essential. Three fundamental steps were chosen, based on the literature [47]:

- Image cropping
- Gray monochrome conversion and image binarization (thresholding)
- Edge Detection

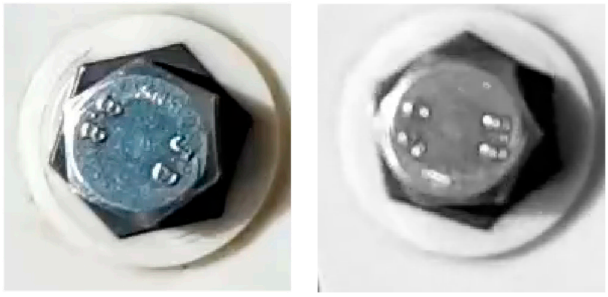
The first step is fundamental for removing the issues related to the background and in particular for improving the computational speed.

The second step is used to prepare the image for edge detection, limiting the effect of noise, illumination changes and occlusions in the search image. In this analysis, in fact, the edges are used as features to enhance the TM. The hybridization of template-based and feature-based approaches is not new (e.g., [48,49]), and allows to overcome the well-known issues of the traditional TM related to scaling and rotations of the search image with respect to the template. This will be the subject of Section 2.4.

2.3.1. Image Cropping

The image cropping is meant to remove the background, solving issues related to changes and clutter in part of the image of little relevance. Furthermore, decreasing the overall number of pixels in the image, the computational burden is reduced.

In this analysis, the image is shrunk from a matrix of 1080 × 1920 px to a matrix of 191 × 191 px, by cropping in the square region defined by the row indices in the range $475 \div 665$ px and the column indices in the range $485 \div 675$ px. The result is presented in Figure 3 for two frames of the video.



(**a**) (**b**)

Figure 3. First (**a**) and thousandth (**b**) frames of the video. Notice how the image (b) is out of focus and the center of the screw is translated upwards. N.B., The selected x coordinate goes from left to right, while the y coordinate goes from the top to the bottom, differently from the standard.

2.3.2. Gray Monochrome Conversion and Image Binarization (Thresholding)

Conversion of an arbitrary color image to grayscale is not a unique procedure in general, as a different weighting of the color channels can effectively represent the effect of shooting black-and-white film. A common strategy is to use the principles of photometry and colorimetry to calculate the grayscale values so as to have the same relative luminance (i.e., the density of luminous intensity per unit area in a given direction) as the original color image. Given the RGB intensities (i.e., values in the range $0 \div 255$ for the RGB24 file format or normalized to $0 \div 1$) provided by the three channels R, G, and B, the luminance Y is defined as a weighted sum of these components. In this analysis, the coefficients from the ITU-R Recommendation BT.601 standard, revision 7 [50] are taken, so as to find:

$$Y = 0{,}299\ R + 0{,}587\ G + 0{,}114\ B \quad (10)$$

The formula reflects the eye color photoreceptors sensitivity, which has a maximum in the green-light region. Notice that, in general, human-perceived luminance is commonly referred to as brightness, while luma is the luminance of an image as displayed by a monitor.

The so obtained grayscale image is displayed in Figure 4a.

Once the gray monochrome image is obtained, thresholding is implemented. The goal of thresholding is to classify pixels as either dark (0) or light (1) to produce a black and white (i.e., binary) image based on the luminance information.

In its simplest implementation, the threshold is a constant set by the user, and the pixels' luminance is compared against this value. An automatic selection of the threshold was implemented by Otsu [51] as a Fisher's Discriminant Analysis performed on the intensity histogram. Otsu's threshold is then determined by minimizing intra-class intensity variance, or equivalently, by maximizing inter-class variance (N.B., the two classes are obviously dark vs. light).

Nevertheless, illumination changes in the image can lead to a bad classification. In this case, an adaptive threshold such as the Bradley's could perform much better [52]. The idea is to use a local threshold which can vary within the image as it is adapted to the average of surrounding pixels. Typically, a moving window of approximately 1/8th of the size of the image is used for computing the local mean intensity. Matlab implementation [52] also allows to tune the threshold using a scalar "sensitivity" in the range $0 \div 1$: high sensitivity value leads the thresholding of more pixels as foreground (i.e., class 1, light), at the risk of including some background pixels (i.e., class 0, dark).

Thresholding, in fact, is commonly used to separate foreground objects from their background, reinforcing the action of the image cropping.

Comparing Figure 4c to 4b, the robustness of Bradley's adaptive thresholding to illumination changes is highlighted. Furthermore, it can be noticed that the circle in the image background is removed, improving the robustness to background changes and clutter.

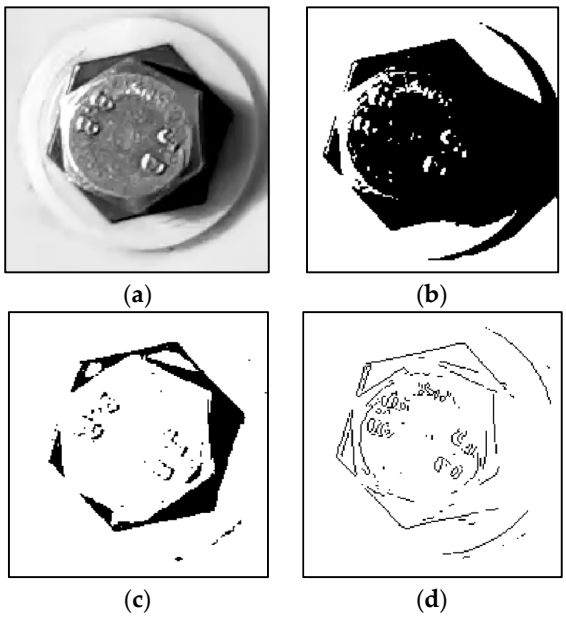


Figure 4. (**a**) Original grayscale: Grayscale conversion of the first frame of the video (Figure 3a) using luminance [50]; (**b**) Otsu threshold: Comparison of Otsu's thresholding [51]; (**c**) Adaptive thresholding (Bradley) [52,53] with sensitivity: 0,99; (**d**) Not-Sobel filtering [54,55] applied to the grayscale image in (a).

2.3.3. Edge Detection

The objective of edge detection is to find the locations in a grayscale image where the change in intensity (i.e., Y) is sufficiently large to be taken as a reliable indication of an edge [35]. One of the most common detectors is the Differential Gradient edge detector called Sobel–Feldman filter [54], which uses two 3 × 3 windows convolved with the image to produce two directional pieces of information (i.e., approximated gradients) added to find the resulting magnitude. Finally, the magnitude information undergoes thresholding to produce a binary image of the edges (automatic heuristic threshold selection [55]), as reported in Figure 4d, where the logical not operator is applied to highlight the edges in black. As can be easily noticed in the picture, the edges are filtered and isolated very effectively, but the illumination affects the result.

By comparing Adaptive thresholding and Sobel filtering (i.e., Figure 4c vs. Figure 4d) it is clear how robustness to illumination changes is important in the analysis. Hence, in this work, the edge detection is left to Bradley's adaptive thresholding, to produce a search image with thicker edges more robust to noise, illumination changes, and background clutter (i.e., Figure 4c).

To summarize, the finally selected preprocessing is reported in Figure 5.

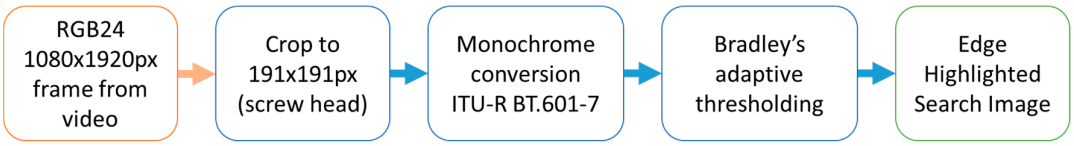


Figure 5. Preprocessing scheme.

2.4. GA-adaptive Template Matching

In traditional TM, as described in Section 2.2, the template is selected as a cutout from one larger reference search image and compared to all the successive test search images using the cross-correlation. The position of maximum correlation testifies the match, proving the template detection. This obviously works very well in case of a fixed framing camera depicting an object which translates on a plane orthogonal to the optical axis of the camera. Nevertheless, in case of rotations, scale changes or non-rigid transformations (N.B., images are a projection of a 3D scene onto a 2D plane, so that movement on a plane non-orthogonal to the optical axis of the camera can lead to deformations), the method cannot be used unless some technical device is implemented, such as using multiple templates with different scales and rotations (e.g., eigenspaces) or using Deformable Part Models (DPM) or implementing Deformable Template Matching [37]. Nevertheless, in this particular work, the Genetic Algorithm was selected to deal with rigid transformations of the template. The GA was used for adapting a parametric template so as to get the maximum correlation (i.e., the best match). The complete cross-correlation function (i.e., correlation for all the delays or lags) is never computed in this case; the GA is exploited for optimizing at the same time not only the scale and the orientation but also the location of the parametric template in the search space, so as to obtain an hybrid of the template-based and feature-based approaches which allows to overcome the issues of the traditional TM (i.e., scaling and rotations).

Notice that, in the SURVISHNO video, the framing is fixed, but the fan revolution occurs in a plane non-perfectly orthogonal to the optical axis, so that, during the revolution, the center of rotation of the fan moves in a small region, while the image undergoes slight deformations. These perspective-related deformations are neglected by the here-introduced algorithm, but relative translations of the locking screw head hexagon and the underlying hexagon (lying on two different planes) are allowed by breaking the template adaptation in three successive GA steps.

2.4.1. Template Parametric Model

The template parametric model arose from the exploitation of the geometrical features of the search image. In particular, three characteristic features were defined in order to determine the angle of rotation of the fan. The first two are related to the regular hexagonal shape of the screw head and the underlying driving shaft. The third is the resistance class logo (i.e., 8.8), which enables to discern the orientation of the screw and consequently that of the fan.

The three characteristic features are highlighted in Figure 6.

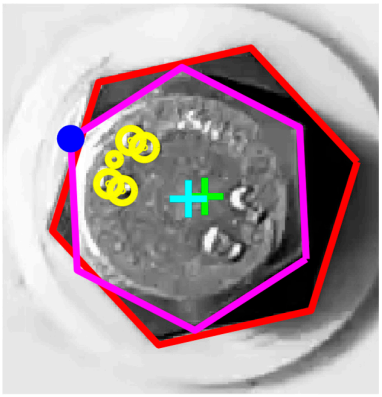


Figure 6. Geometrical edge features highlighted on the first frame of the video (Figure 3a): Outer hexagon model in red and its center in green, inner hexagon model in magenta and its center in cyan, 8.8 logo model in yellow, and the finally tracked vertex in blue.

Two parametric models for the edges are then built. The first represents a hexagon inscribed in a circumference and is governed by the coordinates of the center (i.e., the location), the radius of the circumference in which the hexagon is inscribed (i.e., the scale), and the angle of rotation of the

hexagon. The second is the 8.8 logo, modeled as 5 circles, around one of the diagonals of the screw hexagon, as reported in yellow in Figure 6. This is governed by five parameters: a size parameter ruling the radii of the circles and the height of the writing, a width parameter giving the distance of the two 8 characters, a shift parameter allowing uneven positioning of the two 8 characters around the main axis, a radial distance of the writing from the center of the screw hexagon (either positive or negative to cover both sides of the reference axis with respect to the center of the screw), and an angular deviation from the reference axis, whose information is considered as a known input given the desired diagonal of the screw.

Given these two geometric models, a binary template of size 191 × 191 px can be produced as a function of these 13 parameters plus thickness information. The characteristic parameters are reported in Table 1. The ideal path from the model, defined in the continuous pixel space, can be used as a mask for lighting (i.e., turning to 1) the pixels covered by such a filter. The path thickness is obviously a relevant parameter, but to avoid overcomplicating the model, the thickness was pre-set to a constant value of 30 px for the hexagons (t and t' in Figure 7), while it is related to the scale parameter s for the 8.8 logo ($thickness = r_1 - r_2 = 3{,}5\,s$).

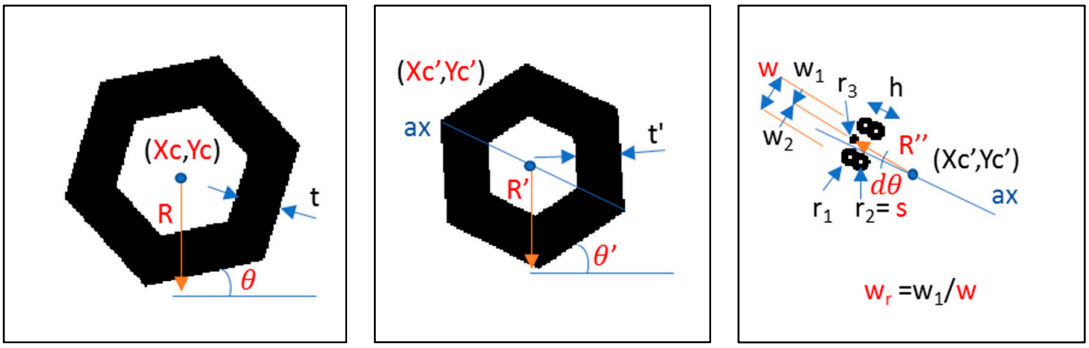


Figure 7. Binary templates after Genetic Algorithm (GA) optimization on the first frame (Figure 3a) with characteristic parameters highlighted in red (overall, 13 independent parameters, as reported in Table 1): (**a**) Outer hexagon template; (**b**) Inner hexagon template; (**c**) 8.8 logo template. N.B., The pictures "quantization" effect is determined by the 191 × 191 px grid of the search image, which dictates the final template resolution.

Table 1. Characteristic variables of the three parametric templates. The 13 independent variables are highlighted in red, t and t' parameters are constants, while the other parameters are derived.

Parameter	*Description*	*Parameter*	*Description*
X_c, Y_c	Center of the outer hexagon (OH)	R''	Distance of 8.8 logo from (X_c', Y_c')
R	Radius of the inscribing circle (OH)	$d\theta$	Deviation from ax slope direction
θ	Rotation of the OH	$s = r_2$	Logo size = hollow circles radii
t	Thickness of OH	r_1	Logo's circles radii $r_1 = 4{,}5s$
X_c', Y_c'	Center of the inner hexagon (IH)	r_3	Logo's dot radius $r_3 = 2{,}25s$
R'	Radius of the inscribing circle (IH)	h	Logo's height $h = 5{,}5s$
θ'	Rotation of the IH	w	Logo's width $w = w_1 + w_2$
t'	Thickness of the IH	w_r	Logo's width ratio $w_r = w_1/w$
ax	8.8 intercepting diagonal of IH	w_1, w_2	Distance of "8" from Logo's dot

As highlighted in Figure 7, three different templates were actually generated, as the template adaptation was performed in three different subsequent GA optimizations, exploiting in the following steps the knowledge acquired from the previous optimization.

In particular, the optimized outer hexagon path is used as a mask for cropping the search image and further remove the background, improving the following GA search. Then, from the optimized inner hexagon, the diagonal on which the 8.8 logo lies is detected (i.e., "ax" in Figure 7b,c, found by

summing the pixels intersecting the three diagonals and seeking the maximum), and the information is used as input for the last GA optimization.

It is important to point out that using a parametric template-mask defined on a continuous search space and implementing a GA optimization of the match between the corresponding discrete template image and the search image, it is possible to obtain a parametric estimation that goes beyond the pixel grid, leading to a super-resolution (i.e., similar to what obtained in [30]).

2.4.2. Objective Function

The change of paradigm from traditional TM to GA-adaptive TM is related to the use of GA for the estimation of the optimal parameters maximizing the match of the parametric template to the search image. In order to evaluate "how good" a reconstructed template is (N.B., reconstructed on the basis of the selected parameters), an objective function (commonly called utility function when referred to maximization problems or cost function when dealing with minimizations) is needed. In order to keep the link between TM and GA-adaptive TM, a possible utility function is the correlation function (i.e., r in Equation (11)). Nevertheless, in the literature, other commonly found functions are the Sum of Absolute Differences (i.e., SAD) or the Sum of Squared Differences (i.e., SSD), usually implemented as cost functions for a minimization problem.

In order to select the best objective function for this particular implementation, two considerations are fundamental. First, in this work the template is reconstructed to the same size ($JxK = 191x191$) of the search image so that all the objective functions can be easily implemented as:

$$r(param) = \sum_{j=1}^{J} \sum_{k=1}^{K} t[j,k|param]\, s[j,k] \tag{11}$$

$$SAD(param) = \sum_{j=1}^{J} \sum_{k=1}^{K} |t[j,k|param] - s[j,k]\,| \tag{12}$$

$$SSD(param) = \sum_{j=1}^{J} \sum_{k=1}^{K} (\,t[j,k|param] - s[j,k])^2 \tag{13}$$

where $t[j,k|param]$ is the reconstructed template as a function of the corresponding parameters (i.e., $param$, see Figure 7) and $s[j,k]$ is the search image (e.g., a frame processed to obtain the result in Figure 4c).

Second, the t and s are binary, so that the possible results for single pixel information can be summarized as in Table 2:

Table 2. Correlation r, Sum of Absolute Differences (SAD) and Sum of Squared Differences (SSD) comparison for binary images.

t	s	→	r	SAD	SSD
1	1		1	0	0
1	0		0	1	1
0	1		0	1	1
0	0		0	0	0

From Table 2, it is clear that SAD and SSD are equivalent in the case of binary images. Another relevant consideration regards the fact that correlation can be used for maximizing the match (i.e., the similarity), while SAD and SSD are suitable for minimization of the mismatch (i.e., the difference). Nevertheless, correlation rewards the similarity of white pixels (i.e., the 1) only, but neglects the black pixels (i.e., the 0). On the contrary, SAD and SSD penalize only the different pixels, or in other words, rewards both the white matching and the black matching pixels. As a result of this, the correlation was used as a utility function for the first GA (so that, thanks to the selected template shape Figure 7a, the 8.8 and JD logos are not accounted in the match), while the SAD was selected as a cost function for the second and third GA optimization.

2.4.3. Genetic Algorithm Optimization

Optimization is the selection of the best element from a set of available alternatives according to some criteria. In a more formal way, given an objective function $f: S \to R$ which links the search space of feasible solutions to the corresponding utility or cost, the optimization process seeks to find the element $x_o \in S$ such that $f(x_o) \leq f(x)\ \forall\, x \in S$ (minimization) or such that $f(x_o) \geq f(x)\ \forall\, x \in S$ (maximization). Fixing a target for convenience, in the simplest case, an optimization problem corresponds to the minimization of a cost function over a search space obtained by constraining the overall Euclidean space. Or, $\underset{x \in S}{\operatorname{argmin}} f(x)$. From a mathematical point of view, the minimization of a function typically involves derivatives. Then, the more a function is complex (e.g., defined on a wide multidimensional support, non-continuous, or with non-continuous derivatives, featuring many local minima, etc.), the harder is the computation of such derivatives, so that the optimization may become very tricky in practical cases. Furthermore, the optimization is very likely to get stuck into local minima in the vicinity of an initial guess value for the optimum location (local optimization), with no guarantees (unless particular properties of the cost function i.e., convexity) that the result corresponds to the actual global minimum (global minimization).

In general, the assessment of the performance of an optimizer can be expressed in terms of:

- Exploration: the optimizer discovers a wide region of the search space,
- Exploitation: the optimizer "pounds the pavement" on a limited but promising region,
- Reliability: repeatability of the fund solution.

It is important to highlight that exploration and exploitation are competing properties. Local optimizers show very good exploitation at the expense of a very poor exploration. On the contrary, a good global optimizer should sacrifice exploitation to gain in exploration and speed. This is usually obtained taking advantage of heuristic or meta-heuristic techniques implementing some form of stochastic optimization.

An important category of global population-based metaheuristic optimization algorithms is the Evolutionary. An evolutionary algorithm (EA) uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection. Candidate solutions to the optimization problem play the role of individuals in a population, and the cost function determines the quality of a solution. A "direct search" is performed to find the best individuals within the population according to their quality. These best individuals are then selected to determine the offspring, namely the new trial solutions, which will substitute lower quality individuals.

The most famous EA is the Genetic Algorithm, developed by John Holland introduced genetic algorithms in 1960 based on the concept of Darwin's theory of evolution. The GA evolutionary cycle starts initiating a population randomly and evaluating the quality of each individual on the basis of his genotype. The best individuals are then selected to produce via modification the new offspring, while the worst are discarded. Modifications are stochastically triggered operators such as the crossover (the offspring is a random mix of the genotypes of their parents) or the mutation (the offspring features new genes which were not present in the parents). The first is important to ensure exploitation, while the second guarantees exploration of the search space of all possible genotypes. Finally, a new population is ready for starting again the cycle until some stopping criteria are met. The cycle is outlined in Figure 8.

The GA selected inputs in this work were:

- Population Size: $N_p = 100$.
- Elite Count: 5%. It defines the number of best individuals selected as a percentage of N_p.
- Crossover Fraction: 80%. It defines the offspring quantity at the next generation as a percentage of N_p. As the total N_p is fixed, the percentage of discarded individuals equals the crossover fraction.
- Default mutation: Shrinking Gaussian. Each newborn features a degree of random mutation which decreases in time according to the linear law: $\sigma_g = \sigma_{g-1}\left(1 - c\,\frac{g}{G}\right)$. Where $\sigma_0 = 1$, $c = 1$, and g is the generation index, increasing with time.
- Stopping criterion: maximum number of generations $G = 40$.

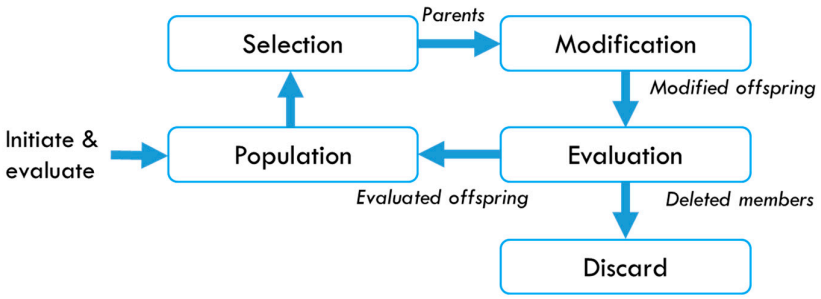


Figure 8. The GA evolutionary cycle.

2.5. Overall Methodology

Finally, the overall methodology to be repeated for all the frames in the video is summarized in the following steps:

1. GA optimization of the outer hexagon template (Figure 7a) to match the search image (e.g., Figure 4c).
    - The outer hexagon path is used to make a mask isolating the foreground of interest and improving the next step.
2. GA optimization of the inner hexagon template (Figure 7b) to match the search image cropped using the outer hexagon path as mask.
    - The inner hexagon path is used to make a mask for isolating the foreground of interest and improving the next step.
    - The three inner hexagon diagonals are tested to find the diagonal around which the 8.8 logo is reported.
3. GA optimization of the 8.8 logo (Figure 7c) to match the search image cropped using the inner hexagon path as a mask.

Thanks to this procedure, all the 13 parameters of interest (Figure 7) can be estimated in all the frames of the video. The position of the 8.8 logo is used to identify a unique vertex of the inner hexagon, from which it is easy to derive the angular position of the screw (and then the angular position of the fan) over time.

2.5.1. IAS Estimation

Once the 13 parameters of interest are available over time, several different IAS estimations can be obtained. In this work, two methodologies are compared.

The first and simplest consists in differencing the angle over time and rescaling over the time interval determined by the sampling frequency (i.e., $30\ fps \rightarrow f_s = 30\ Hz$). Being $\alpha(n)$ is the angle of the vertex identified by the 8.8, and $t = n\,\Delta t = n/f_s$, it is easy to write:

$$\omega(t) = \frac{d\alpha}{dt} \cong \frac{\Delta\alpha}{\Delta t} \quad [rad/s] \tag{14}$$

For this to be accurate, the angle signal should first be "unwrapped" (i.e., corrected by adding multiples of $\pm 2\pi$ when absolute jumps between consecutive elements are greater than π radians). Furthermore, a perfect recognition of all the three templates is required. Nevertheless, even if the match for two hexagons is good for all the frames, it is not the same for the 8.8 logo, which, in some cases (often at a determined angle because of illumination issues), is confounded with the manufacturer logo "JD" (e.g., see Figure 6). In this case, an error of about $\pm\pi$ radians should be compensated in a pre-processing stage.

The second method is based on phase demodulation via the Hilbert analytic signal of a fan speed-related harmonic [7,56]. In this particular case, it was noticed that the signal corresponding to the coordinates of the center of the outer hexagon (i.e., X_c, Y_c in Figure 7a) features a speed-related harmonic because of the not perfect alignment of the optical axis of the camera with the fan axis, leading to a circular movement of the tracked center point.

In a mathematical framework:

1. The analytic signal is computed via Hilbert transform
$$\alpha_{an}(t) = x(t) + iy(t) = A(n)e^{i\Phi(t)} \tag{15}$$
2. The instantaneous frequency is defined as
$$f(t) = \frac{1}{2\pi}\frac{d\Phi(t)}{dt} \quad [Hz] \tag{16}$$
3. From which, a more suitable discrete-time ($t = n\,\Delta t$) implementation can be derived [56]
$$f(n) = \frac{fs}{2\pi}\text{atan}\left(\frac{x(n)y(n+1) - x(n+1)y(n)}{x(n)x(n+1) + y(n)y(n+1)}\right) \tag{17}$$

Despite the "unwrap" and the error compensation, the IAS estimated from the two methods could still be noisy, with large observable discontinuities which are unphysical and incompatible with the principle "Natura non facit saltus", so that a smoothing could be required.

Both the methods were tested on the data to validate the two IAS estimates.

3. Results

The application of the here described GA-adaptive TM led to the estimation of a set of 13 parameters fully specifying the selected template model over time (i.e., frame after frame).

As an example, the first two parameters obtained after the first GA step are the coordinates of the outer hexagon in the search space. These are reported in Figure 9. As can be seen from the picture, the X_c coordinate shows some jumps. These can be put in relation with the autofocus. In the same way, it is easy to notice that the coordinate Y_c is subject to a variation of the first-order statistics (i.e., the mean value) over time. This is highlighted by the estimation of the moving average (order: 50 samples) reported in yellow in the graph.

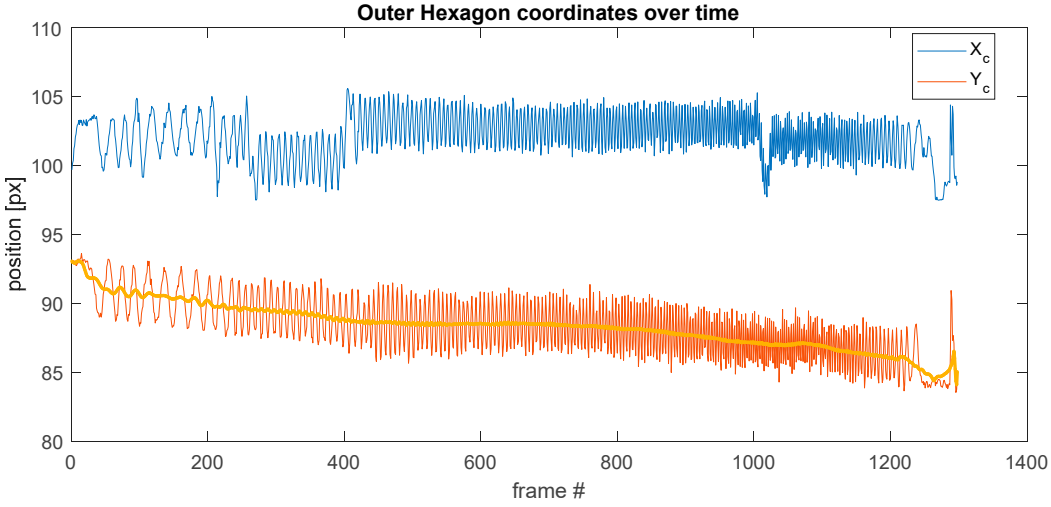


Figure 9. Outer hexagon coordinates over time (frame index). The moving average of the estimated parameter $\boldsymbol{Y_c}$ is added in yellow. N.B., The selected $\boldsymbol{y}$ coordinate ranges from the top to the bottom of the displayed image (e.g., in Figure 6), differently from the standard.

Similar graphs are available for all the 13 optimized parameters. The position of the inner hexagon center and the 8.8 logo vertex were then used to compute the angular position of the fan over time. As can be seen in Figure 10, the angle increments over time are quite uniform if the green, red and magenta points are not considered. In order to get a more consistent estimate, these points should be compensated. As introduced, the green points (360° error) can be avoided by performing and "unwrap" of the angle signal, while the red and magenta points correspond to a wrong localization of the 8.8 logo, which is confounded with the JD logo, so that the 180° error can be distinguished and compensated.

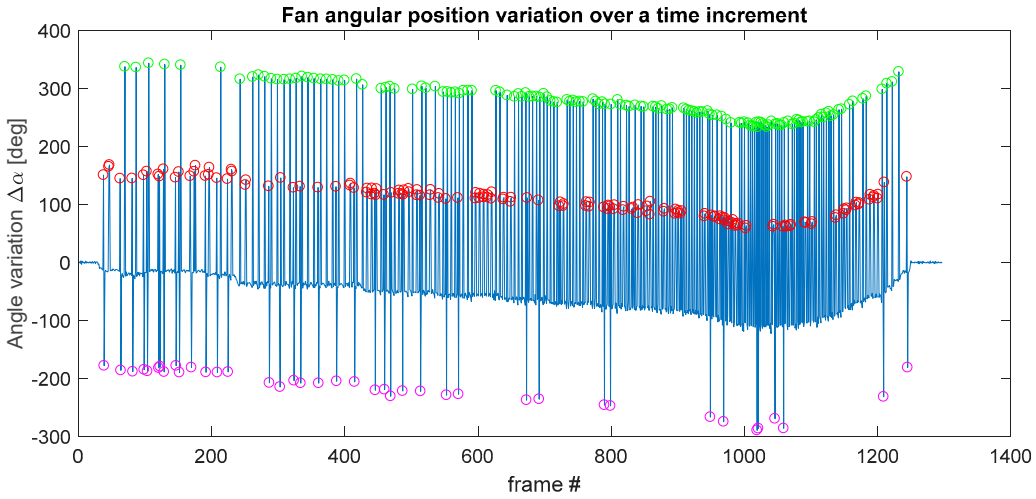


Figure 10. Fan angular position variation (angular increments $\boldsymbol{\Delta\alpha}$) over time (frame index). In green are the points for the "unwrap", in red and magenta are the errors to be compensated (8.8 logo missed).

From the compensated angular increments over time, $\Delta\alpha$, it is easy to obtain the IAS estimate by normalizing this signal with respect to the constant $\Delta t = 1/f_s$, as indicated in Equation (14).

For the sake of comparison, Hilbert phase demodulation was performed on the running-mean-removed Y_c signal so as to produce a second IAS estimate (Equation (17)). The two estimates were then lowpass filtered to produce a more physically reasonable result. The IAS signals after a FIR1 lowpass filter (order: 50 samples, cutoff: 0,1 $f_s/2$) is reported in Figure 11.

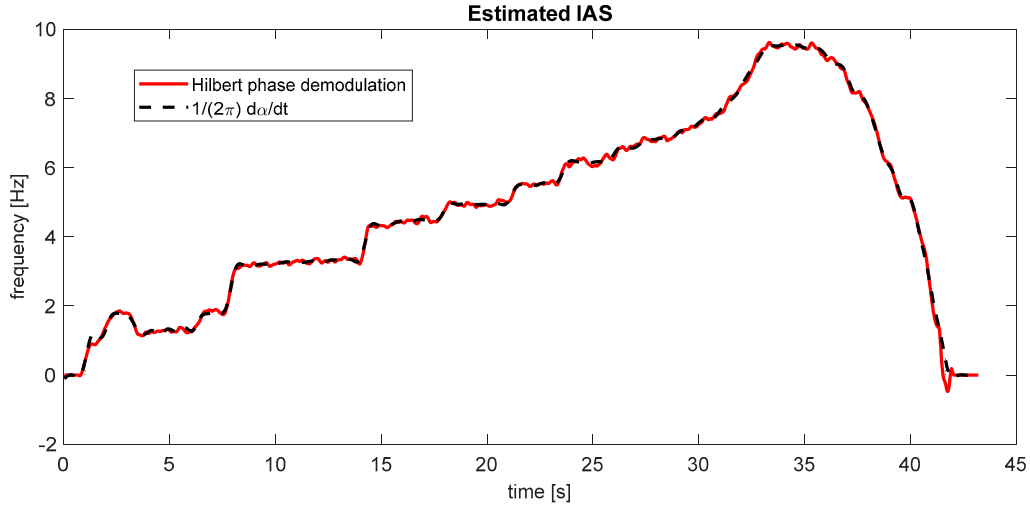


Figure 11. Final Instantaneous Angular Speed (IAS) estimates from the SURVISHNO video.

As can be noticed in Figure 11, the two methods lead to overlapping results. This, then, increases the confidence in the reliability and accuracy of the proposed video-tachometer procedure for the IAS estimation.

4. Discussion and Conclusions

The paper presented a novel method for implementing a cost-effective video-tachometer through a GA-adaptive Template Matching. The target was an offline implementation, as the proposed algorithm is not optimized enough and results slow if running on nowadays PCs. To give an idea, the software execution takes, per each frame, about 2s for the first GA, 2s for the second GA, and 10s for the third GA when using MATLAB R2018b on a machine with 8 GB of ram and an INTEL

i7-7700 CPU at 3,60 GHz. Clearly, for obtaining just the Hilbert demodulation estimate of the IAS, only the first optimization step is needed, so that the computational burden can be strongly reduced, but it is not enough for real-time implementation.

In any case, the method proved to be effective in estimating the IAS of the fan despite the limits of the SURVISHNO 2019 video, acquired using a mobile phone. In fact, the wise selection of the search space effectively dealt with spatial aliasing (i.e., the rolling shutter effect), temporal aliasing (i.e., because of the 10 equal blades of the fan combined with a 30-fps sampling rate), and the additional autofocus distortions. Furthermore, the GA-adaptive implementation of TM reconstructing the binary edge template from a geometrical parametric model demonstrated its robustness to illumination changes and noise in general, as well as to rigid and non-rigid transformations. The issue of background changes and clutter was also tackled both in a pre-processing stage, by cropping and binarizing the search frame, and in the three-step GA optimization, exploiting the information from the previous stages for further focusing the TM on a smaller region of interest.

The here described GA-adaptive TM has the great advantage of allowing the localization of the template with a super-resolution that goes beyond the pixel-grid discretization. This allowed obtaining a robust and reliable estimate of the IAS, avoiding the need of expensive high-resolution encoders or tachometers, which are otherwise necessary for non-stationary machine diagnostics. When the speed is variable, in fact, the machine signature can only be highlighted by resampling the signal synchronously with the angular position of a reference shaft (i.e., performing the so-called Computed Order Tracking to get to the order domain). This way, the events which are phase-locked to the reference shaft (e.g., the intake of a 4 strokes diesel engine to the crankshaft, or the meshing of a broken tooth of a gearwheel to the supporting shaft, etc.) are put in evidence. Furthermore, the IAS is considered a precious diagnostic information per se, so that the analysis of IAS anomalies is spreading in the field of condition monitoring.

To conclude, given the here underlined strengths, the proposed signal processing gives an effective and reliable tool able to foster the IAS-based condition monitoring, setting the state of the art for video-tachometric acquisitions.

Supplementary Materials: The video was downloaded online during the international conference SURVISHNO 2019 at https://survishno.sciencesconf.org/resource/page/id/18.

Author Contributions: Conceptualization, A.P.D.; data curation, A.P.D.; formal analysis, A.P.D.; funding acquisition, L.G.; investigation, A.P.D.; methodology, A.P.D.; project administration, L.G.; software, A.P.D.; supervision, Luigi Garibaldi; validation, Luigi Garibaldi; visualization, A.P.D.; writing—original draft, A.P.D.; writing—review and editing, A.P.D.

Funding: This research received no external funding.

Acknowledgments: We thank the SURVISNO 2019 organization committee and in particular Quentin LECLERE, Hugo ANDRE and Emmanuel FOLTETE for acquiring the video and organizing the challenge.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lin, J.; Zhao, M. A review and strategy for the diagnosis of speed-varying machinery. In Proceedings of the 2014 International Conference on Prognostics and Health Management, Cheney, WA, USA, 22–25 June 2014.
2. Fyfe, K.R.; Munck, E.D.S. Analysis of computed order tracking. *MSSP* **1997**. doi:10.1006/mssp.1996.0056.
3. Brandt, A.; Lago, T.; Ahlin, K.; Tuma, J. Main principles and limitations of current order tracking methods. *Sound Vib.* **2005**, *39*, 19–22.
4. Bonnardot, F.; El Badaoui, M.; Randall, R.B.; Daniere, J.; Guillet, F. Use of the acceleration signal of a gearbox in order to perform angular resampling (with limited speed fluctuation). *MSSP* **2005**. doi:10.1016/j.ymssp.2004.05.001.
5. Coats, M.; Sawalhi, N.; Randall, R.B. Extraction of tacho information from a vibration signal for improved synchronous averaging. *Acoustics* **2009**.

6. Remond, D.; Antoni, J.; Randall, R.B. Instantaneous Angular Speed (IAS) processing and related angular applications. *MSSP* **2014**, *45*, 24–27. doi:10.1016/j.ymssp.2013.10.015.
7. Peeters, C.; Leclère, Q.; Antoni, J.; Lindahl, P.; Donnal, J.; Leeb, S.; Helsen, J. Review and comparison of tacholess instantaneous speed estimation methods on experimental vibration data. *MSSP* **2019**. doi:10.1016/j.ymssp.2019.02.031.
8. Antoni, J.; Griffaton, J.; André, H.; Avendaño-Valencia, L.D.; Bonnardot, F.; Cardona-Morales, O.; Castellanos-Dominguez, G.; Daga, A.P.; Leclère, Q.; Vicuña, C.M.; et al. Feedback on the surveillance 8 challenge: vibration-based diagnosis of a safran aircraft engine. *MSSP* **2017**. doi:10.1016/j.ymssp.2017.01.037.
9. Ben Sasi, A.Y.; Gu, F.; Payne, B.; Ball, A. Instantaneous angular speed monitoring of electric motors. *J. Qual. Maint. Eng.* **2004**. doi:10.1108/13552510410539196.
10. Lebaroud, A.; Clerc, G. Diagnosis of Induction Motor Faults Using Instantaneous Frequency Signature Analysis. *ICEM* **2008**. doi:10.1109/ICELMACH.2008.4799993.
11. Spagnol, M.; Bregant, L.; Boscarol, A. Electrical Induction Motor Higher Harmonics Analysis Based on Instantaneous Angular Speed Measurement. In Proceedings of the Fourth International Conference on Condition Monitoring of Machinery in Non-Stationary Operations, CMMNO'2014, Lyon, France 15–17 December 2014. doi:10.1007/978-3-319-20463-5_3.
12. Yang, J.; Pu, L.; Wang, Z.; Zhou, Y.; Yan, X. Fault detection in a diesel engine by analysing the instantaneous angular speed. *MSSP* **2001**. doi:10.1006/mssp.2000.1344.
13. Desbazeille, M.; Randall, R.B.; Guillet, F.; El Badaoui, M.; Hoisnard, C. Model-based diagnosis of large diesel engines based on angular speed variations of the crankshaft. *MSSP* **2010**. doi:10.1016/j.ymssp.2009.12.004.
14. Chang, Y.; Hu, Y. Monitoring and fault diagnosis system for the diesel engine based on instantaneous speed. *ICCAE conference* **2010**. doi:10.1109/ICCAE.2010.5451800.
15. Antonopoulos, A.K.; Hountalas, D.T. Effect of instantaneous rotational speed on the analysis of measured diesel engine cylinder pressure data. *Energy Convers Manag.* **2012**. doi: 10.1016/j.enconman.2012.01.020.
16. Lin, T.; Tan, A.; Ma, L.; Mathew, J. Condition monitoring and fault diagnosis of diesel engines using instantaneous angular speed analysis. *Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci.* **2014**. doi:10.1177/0954406214533253.
17. Liu, Y.; Gu, L.; Yang, B.; Wang, S.; Yuan, H. A new evaluation method on hydraulic system using the instantaneous speed fluctuation of hydraulic motor. *Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci.* **2018**. doi:10.1177/0954406217722576.
18. Roy, S.K.; Mohanty, A.R.; Kumar, C.S. Fault detection in a multistage gearbox by time synchronous averaging of the instantaneous angular speed. *J. Vib. Control* **2016**. doi:10.1177/1077546314533582.
19. Li, B.; Zhang, X. A new strategy of instantaneous angular speed extraction and its application to multistage gearbox fault diagnosis. *J. Sound Vib.* **2017**. doi:10.1016/j.jsv.2017.02.043.
20. Zhao, M.; Jia, X.; Lin, J.; Lei, Y.; Lee, J. Instantaneous speed jitter detection via encoder signal and its application for the diagnosis of planetary gearbox. *MSSP* **2018**. doi:10.1016/j.ymssp.2017.04.033.
21. Liang, L.; Liu, F.; Kong, X.; Li, M.; Xu, G. Application of Instantaneous Rotational Speed to Detect Gearbox Faults Based on Double Encoders. *Chin. J. Mech. Eng.* **2019**. doi:10.1186/s10033-019-0324-z.
22. Zimroz, R.; Urbanek, J.; Barszcz, T.; Bartelmus, W.; Millioz, F.; Martin, N. Measurement of Instantaneous Shaft Speed by Advanced Vibration Signal Processing—Application to Wind Turbine Gearbox. *Metrol. Meas. Syst.* **2011**. doi:10.2478/v10178-011-0066-4.
23. Renaudin, L.; Bonnardot, F.; Musy, O.; Doray, J.B.; Rémond, D. Natural roller bearing fault detection by angular measurement of true instantaneous angular speed. *MSSP* **2010**. doi:10.1016/j.ymssp.2010.05.005.
24. Bourdon, A.; André, H.; Rémond, D. Introducing angularly periodic disturbances in dynamic models of rotating systems under non-stationary conditions. *MSSP* **2014**. doi:10.1016/j.ymssp.2013.09.004.
25. Lamraoui, M.; Thomas, M.; El Badaoui, M.; Girardin, F. Indicators for monitoring chatter in milling based on instantaneous angular speeds. *MSSP* **2014**. doi:10.1016/j.ymssp.2013.05.002.
26. Leclere, Q.; André, H.; Antoni, J. A multi-order probabilistic approach for Instantaneous Angular Speed tracking. debriefing of the CMMNO'14 diagnosis contest. *MSSP* **2016**. doi:10.1016/j.ymssp.2016.02.053.
27. Li, Y.; Gu, F.; Harris, G.; Ball, A.; Bennett, N.; Travis, K. The measurement of instantaneous angular speed. *MSSP* **2005**. doi: 10.1016/j.ymssp.2004.04.003.

28. Sushma Sri, M. A study on motion estimation. *Int. J. Mod. Trends Eng. Res. (IJMTER)* **2017**. doi: 10.21884/IJMTER.2017.4188.7PJJ9.
29. Torr, P.H.S.; Zisserman, A. Feature Based Methods for Structure and Motion Estimation. *ICCV Workshop Vis. Algorithms* **1999**. doi:10.1007/3-540-44480-7_19.
30. Irani, M.; Anandan, P. About Direct Methods, *ICCV Workshop Vis. Algorithms* **1999**. doi:10.1007/3-540-44480-7_18.
31. Egmont-Petersen, M.; de Ridder, D.; Handels, H. Image processing with neural networks—A review. *Pattern Recognit.* **2002**, 35, 2279–2301. doi:10.1016/S0031-3203(01)00178-9.
32. Tolba, A.; El-Baz, A.; El-Harby, A. Face Recognition: A Literature Review. *Int. J. Signal Process.* **2005**, *2*, 88–103.
33. Solanki, K.; Pittalia, P. Review of Face Recognition Techniques. *Int. J. Comput. Appl.* **2016**, *133*, 20–24. doi:10.5120/ijca2016907994.
34. Brunelli, R. Template Matching Techniques in Computer Vision: Theory and Practice; Wiley: Hoboken, NJ, USA, 2009.
35. Leavers, V.F. Computer Vision: Shape Detection. In *Shape Detection in Computer Vision Using the Hough Transform;* Springer, London, UK, **1992**. doi: 10.1007/978-1-4471-1940-1_1.
36. Talmi, I.; Mechrez, R.; Zelnik-Manor, L. Template Matching with Deformable Diversity Similarity. Available online: http://openaccess.thecvf.com/content_cvpr_2017/html/Talmi_Template_Matching_With_CVPR_2017_paper.html (accessed on 20 December 2019).
37. Lee, H.; Kwon, H.; Robinson, R.M.; Nothwang, W.D. DTM: Deformable Template Matching. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016. arXiv:1604.03518.
38. Muramatsu, S. Strategy of high-speed template matching and its optimization by using GA. *Syst. Comput. Jpn.* **2003**. doi:10.1002/scj.1192.
39. Karungaru, S.; Fukumi, M.; Akamatsu, N. Face recognition using genetic algorithm based template matching, *IEEE Int. Symp. Commun. Inf. Technol.* **2004**. doi:10.1109/ISCIT.2004.1413920
40. Velasco, H.M.G.; Orellana, C.J.G.; Macías, M.M.; Caballero, R.G.; Sotoca, M.I.A. Application of Repeated GA to Deformable Template Matching in Cattle Images. In Proceedings of the Third International Workshop, AMDO 2004, Palma de Mallorca, Spain, 22–24 September 2004. doi:10.1007/978-3-540-30074-8_14.
41. Zhang, X.; Liu, Q.; Li, M. A Template Matching Method Based on Genetic Algorithm Optimization. *Appl. Mech. Mater.* **2012**. doi: 10.4028/www.scientific.net/AMM.220-223.1298.
42. El-Baz, A.; Elnakib, A.; Abou-El-Ghar, M.; Gimel'farb, G.; Falk, R.; Farag, A. Automatic Detection of 2D and 3D Lung Nodules in Chest Spiral CT Scans. *Int. J. Biomed. Imaging* **2013**. doi:517632. 10.1155/2013/517632.
43. Kay, S.; Rangaswamy, M. The Ubiquitous Matched Filter: A Tutorial and Application to Radar Detection. In *Classical, Semi-classical and Quantum Noise*, Cohen, L., Poor, H., Scully, M., Eds; Springer: New York, NY, USA, 2012. doi:10.1007/978-1-4419-6624-7_8.
44. Turin G. L. An introduction to matched filters. *IRE Trans. Inf. Theory* **1960**, 6, 311–329. doi:10.1109/TIT.1960.1057571.
45. Vleck, J.V.; Middleton, D. A theoretical comparison of the visual, aural, and meter reception of pulsed signals in the presence of noise. *J. Appl. Phys.* **1946**, *17*, 940–971. doi:10.1063/1.1707666.
46. North, D.O. An analysis of the factors which determine signal/noise discrimination in pulsed-carrier systems. *Proc. IEEE* **1943**, *51*, 1016–1027. doi:10.1109/PROC.1963.2383.
47. Mohd, F.Z.; Hoo, S.C.; Shahrel, A.S. Object Shape Recognition in Image for Machine Vision Application. *Int. J. Comput. Theory Eng.* **2012**, 76–80. doi:10.7763/IJCTE.2012.V4.428.
48. Kumar, V.; Pandey, S.; Pal, A.; Sharma, S. Edge Detection Based Shape Identification. Available online: https://arxiv.org/ftp/arxiv/papers/1604/1604.02030.pdf (accessed on 20 December 2019).
49. Chai, T.Y.; Rizon, M.; San, W.; Seong, T. Facial Features for Template Matching Based Face Recognition. *Am. J. Appl. Sci.* **2009**. doi:10.3844/ajassp.2009.1897.1901.
50. ITU-R recommendation BT.601 standard, revision 7, 2011. Available online: https://www.itu.int/rec/R-REC-BT.601/en (accessed on 20 December 2019).
51. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. doi:10.1109/TSMC.1979.4310076.

52. Bradley, D.; Roth, G. Adaptive Thresholding using the Integral Image. *J. Graph. Tools* **2007**. doi:10.1080/2151237X.2007.10129236.
53. MATLAB Matworks "imbinarize" function. Available online: https://www.mathworks.com/help/images/ref/imbinarize.html (accessed on 20 December 2019).
54. Sobel, I.; Feldman, G. An Isotropic 3x3 Image Gradient Operator. Available online: https://www.researchgate.net/publication/239398674_An_Isotropic_3x3_Image_Gradient_Operator (accessed on 20 December 2019).
55. MATLAB Matworks "edge" function. Available online: https://it.mathworks.com/help/images/ref/edge.html (accessed on 20 December 2019).
56. Barnes, A.E. The calculation of instantaneous frequency and instantaneous bandwidth. *GEOPHYSICS* **1992**, *57*, 1396–1524. doi:10.1190/1.1443220.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).