



Contents lists available at ScienceDirect

CIRP Journal of Manufacturing Science and Technology

journal homepage: www.elsevier.com/locate/cirpj

Vision-based modal analysis of cutting tools

Pulkit Gupta, Harsh Singh Rajput, Mohit Law*

Machine Tool Dynamics Laboratory, Department of Mechanical Engineering, Indian Institute of Technology Kanpur, Kanpur 208016, India

ARTICLE INFO

Article history:
Available online xxx

Keywords:
Dynamics
Cutting tool
Computer vision
Vibration
Image processing
Digital image correlation

ABSTRACT

This paper presents the use of vision-based methods for cutting tool motion registration and modal analysis. Motion of three illustrative tools were recorded using low- and high-speed cameras with sufficiently high resolutions. The tool's own features are used to register motion. Pixels within images from recordings of the vibrating tools are treated as non-contact motion sensors. Comparative analysis of three different methods of motion registration are presented to evaluate their suitability for the application of interest. These include variants of expanded edge detection and tracking schemes, expanded optical flow-based schemes, and established digital image correlation methods. Performance of different methods was observed to be governed by the tool's own features, illumination conditions, noise, and the image acquisition parameters. Extracted motion was benchmarked against twice integrated measured tool point accelerations, and motion was generally observed to compare well. Modal parameters extracted from vision-based measurements were also observed to agree with those extracted using more traditional experimental modal analysis procedures using a contact type accelerometer as the transducer. Since methods presented are generalized, they can suitably be adapted for other applications of interest.

© 2020 CIRP.

Introduction

Cutting tool dynamics are usually evaluated using experimental modal analysis procedures. Since these dynamics govern the chatter vibration-free machining capability of the machine tool system, and since chatter vibrations have no redeeming qualities, accurately measuring the dynamics to mitigate these vibrations is paramount. This paper proposes the use of vision-based non-contact vibration measurement methods for their use in modal analysis of cutting tools.

By treating pixels within images from recordings of the vibrating tool as non-contact motion sensors, and by applying image processing and computer vision techniques, we present methods to register tool motion. The resulting pixel-displacement time series data is then used to extract the modal parameters of interest. Since vision-based measurements are non-contact in nature, the pitfalls associated with mass loading of contact type sensors influencing the dynamics of the measured cutting tool system are avoided. Furthermore, since vision-based measurements allow full-field displacement measurements, mode shape analysis is easier than the experimental modal analysis methods using roving actuators/hammer and/or sensors. Moreover, measurements need just a camera, a lens sometimes, and a computer

with simple computing tools to post-process the recorded video. These advantages make vision-based vibration measurement methods easier to instrument than other non-contact methods such as those using expensive laser vibrometers and data acquisition systems.

Because vision-based vibration measurement offers advantages, these methods have already found favour in civil structural health monitoring of buildings, bridges, and cables [1–3]. Other applications include the dynamic testing of beams, pipes, and plates [4–7], and in the dynamic measurement of wind turbines and helicopters [8,9]. Despite these exemplary examples, and despite the successful use of vision techniques in other engineering disciplines [10], the use of vision-based measurement methods in machine tools has focused more on classification problems. These include classifying good tools from bad [11,12], classifying bad surface quality from good [13,14], including classifying of surfaces with chatter [15–17]. High-speed imaging techniques have also been used to investigate chip formation mechanisms in machining [18–20]. Such uses of vision-based methods in machine tool applications have mostly relied on processing of still images. In contrast, our recent work defines visual vibrometry as a video camera-based vibration measuring technique for machine and cutting tools [21], and as such, that work may be considered an introduction to this one.

Since vision-based vibration measurement methods offer significant advantages, visual vibrometry has the potential to

* Corresponding author.
E-mail address: mLaw@iitk.ac.in (M. Law).

offer an effective alternate paradigm for the modal analysis of cutting tools. To help realize this potential, it is necessary to evaluate the efficacy of different motion registration methods being influenced by the tool's own features, illumination conditions, noise, and the image acquisition parameters. Such a comparative analysis is the main aim of this paper, and is also its key technical contribution.

Methods of motion registration are broadly classified as those using point tracking methods [9], those using template matching schemes based on the principles of digital image correlation (DIC) [7,9,10], and those which use the object's own features to detect and track the vibrating edges to extract their motion [9,21]. Of these three, since point tracking methods require optical markers/targets to be placed on the vibrating structure, and since cutting tools are generally not large enough to place such targets, point tracking methods are less suited to the application of interest in this paper. This paper will instead discuss the use of standard DIC methods [22,23], along with expanded optical flow-based methods that are a complementary variant of the DIC methods [6,10,24–26], and expanded variants of edge detection and tracking methods [27–31].

DIC methods estimate motion by correlating between successive images the position of a series of small and statistically distinctive regions on the surface of the object from within a stochastic speckle pattern made on that surface. Though visual vibrometry using DIC techniques has been used to estimate the in- and out-of-plane motion of large civil infrastructure and other objects [7,9], its use has not yet been extended for estimating cutting tool vibrations – which this paper will address. Since creating speckle patterns on cutting tools may damage coatings on the tools and render them useless, this paper will discuss using the cutting tool's own features to estimate motion. And, instead of using commercial and/or open-source DIC codes that are suited for objects with speckle patterns on them, we implement DIC using established techniques [22,23].

Since DIC analysis includes only spatial information for estimating displacement, this paper also presents the use of optical flow-based methods that incorporate spatial and temporal data to estimate motion [24]. The optical flow-based schemes that we implement is based on the classical work reported in [25,26] that assume brightness constancy between successive frames to estimate displacements in the directions of the intensity gradients only [6,10]. Since gradient-based optical flow is also sensitive to method of evaluating the gradient, and to image noise, this paper presents expanded analysis to understand the influence of these parameters on the response. We also systematically investigate how the efficacy of optical flow methods depend on the region of interest around the vibrating edge(s). Optical flow-based methods implemented herein also use the tool's own features to estimate motion.

Building on the successful use of the Canny edge detector [30] in our earlier reported work to detect and track motion of a machine and different cutting tools using the object's own features [21], this paper will present expanded analysis to compare how other classical methods of edge detection [27–29,31] compare with the Canny edge detection scheme [30]. Expanded analysis presented herein will also consider the sensitivity of the edge detection schemes to image noise.

This paper demonstrates vision-based modal analysis procedures with three representative cutting tools. One example is that of a slender grooving blade, another is that of a slender boring bar, and the third is that of an end mill. Since the grooving blade and the boring bar vibrate at relatively lower frequencies than the end mill, we leverage the capabilities of a typical inexpensive modern smartphone camera [32] to record high megapixel (MP) videos at frame rates of 480 frames per second (fps). To measure higher

frequency vibrations of the boring bar and of the end mill, we use a high-speed camera [33] that records video at 5400 fps with sufficiently high image resolutions. Image acquisition always respects the Nyquist criterion. Setups are detailed in the section describing the experiments.

Methods to extract motion from video recordings of the vibrating tools are outlined in third main section of the paper. These include detailed procedures for the edge detection and tracking schemes, procedures for the expanded optical flow-based scheme, and procedures for DIC. The section on 'comparative analysis of response' benchmarks response extracted using vision-based techniques against twice integrated accelerometer measurements that are measured at the tool point. That section also discusses how results are sensitive to image noise, and to the method of motion registration. Such comparative analysis of different methods of estimating motion of cutting tools presented in this paper is new.

All analyses presented herein are for non-rotating tools and, for when machines are stationary. Furthermore, since mode shape analysis of cutting tools using visual vibrometry has already been addressed in Ref. [21] – this paper limits itself to contrasting only tool point response. Modal parameters thus extracted from the tool point response obtained using vision-based methods are contrasted in the section on 'comparative analysis of modal parameters' with results obtained from conventional experimental modal analysis procedures.

Since vision-based measurements are generally of the output-only type, we prefer to use the robust time-domain Eigensystem Realization Algorithm (ERA) [34] to estimate modal parameters from the output-only data. Even though we prefer ERA, other modal parameter extraction methods [35] may work as well. Since the input force remains unmeasured, the eigenvectors that are estimated from the ERA remain unscaled. And, as such, unscaled eigenvectors cannot be used to reconstruct tool point dynamics characterized by frequency response functions (FRFs) – that necessitates the use of mass-normalized eigenvectors. Eigenvectors can be scaled using two methods. The first involves synchronization of the trigger for image acquisition with the input force measurement. This however is difficult. The other method involves the use of output-only mass-change methods [36], which was already addressed in our previously reported work [21]. Hence, in this paper, to properly scale the FRFs we instead use the scaled eigenvectors estimated from conventional experimental modal analysis procedures in which the input was measured. This is like what was also done by other researchers in Ref. [37]. Results and their discussions in the section on 'comparative analysis of modal parameters' is followed by the main conclusions.

Experimental setups for vision-based vibration measurements

The experimental setup for vision-based vibration measurements of the slender grooving blade is shown in Fig. 1(a). The setup for the end mill is shown in Fig. 1(b), and the setup for the slender boring bar is shown in Fig. 1(c). Tools in all cases were excited with a modal hammer (DYTRAN 5800B4) seen in Fig. 1(a) and the response was recorded using appropriate cameras. Since the displacement signal is only well defined at edges in the video and only in the direction perpendicular to the edges, the single camera setups shown in Fig. 1 can only measure the in-plane motion of the tools. Moreover, since tool motion is expected to be small, image acquisition is planned such as that the pixel resolution can register tool motion.

Separate experiments were also undertaken by mounting a single-axis accelerometer (DYTRAN 322F1) at the tool tip of all tools. The input force and the output accelerations were recorded using National Instruments' (NI) data acquisition system (NI9234

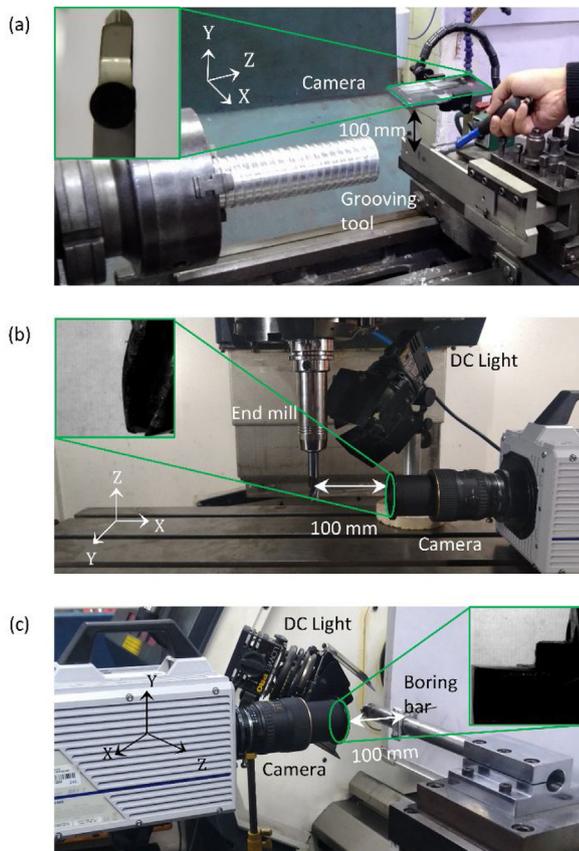


Fig. 1. Experimental setups for vision-based measurements, (a) smartphone camera setup for a grooving tool, (b) high-speed camera setup for an end mill, (c) high-speed camera setup for a boring bar.

+ NI9171) using CutPro[®]'s modal analysis module [38]. Force and acceleration data were sampled at 25.6 kHz.

For the grooving blade measurements, a blade of width 7.5 mm and an overhang of 200 mm was secured in a holder mounted on the bed of a manual lathe machine. The blade was excited in the Z-direction, and resulting in-plane motion (in the X–Z plane) of the blade was recorded with a commercially available smartphone camera (OnePlus 7T) [32] sampling at 480 frames per second (fps) with a pixel resolution of 720×1280 . The smartphone camera used does not have the flexibility to sample with different frame rates and image resolutions. The blade overhang was chosen such that the dominant mode can be captured by the maximum frame rate available on the camera. The distance between the camera and the tool was ~ 100 mm, and the field of view was $27 \text{ mm} \times 48 \text{ mm}$ resulting in a per-pixel resolution of $37.5 \mu\text{m}$.

For measuring the dynamics of the end mill, a regular four-fluted 16 mm diameter end mill was held in a hydraulic expansion toolholder mounted in the spindle of a three-axis CNC milling machine. Tool overhang from the spindle nose was 225 mm. Since end mills typically vibrate at higher frequencies, its motion was recorded using a high-speed camera (Photron FASTCAM SA 1.1) [33] sampling at 5400 fps with a pixel resolution of 1024×1024 . The high-speed camera we used trades image resolution for frame rate. The distance from the lens to the tool was maintained to be 100 mm, and the field of view was $20 \text{ mm} \times 20 \text{ mm}$, resulting in a $20 \mu\text{m}$ pixel resolution. The tool was excited in the Y-direction, and the resulting motion in the Y–Z plane was recorded. For measurements in orthogonal directions, separate setups are necessary [21].

Response for the slender boring bar with a diameter of 25 mm and slenderness ratio of 12.5 were recorded using the smartphone camera as well as separately using the high-speed camera to facilitate comparative analysis. Fig. 1(c) only shows the high-speed camera setup. The tool was excited in the Y-direction, and the resulting in-plane (in the Y–Z plane) motion was recorded. For smartphone-based measurements, the camera recorded video at the rate of 480 fps with a pixel resolution of 720×1280 . The smartphone camera was placed 200 mm away from the tool, resulting in a field of view of $36 \text{ mm} \times 64 \text{ mm}$ with a per pixel resolution of $50 \mu\text{m}$. For the high-speed camera measurements sampling at 5400 fps with a pixel resolution of 1024×1024 , the distance between the lens and the boring bar was maintained to be 100 mm, and, like in the case of the recordings of the end mill, in this case too, the field of view was $20 \text{ mm} \times 20 \text{ mm}$, resulting in a $20 \mu\text{m}$ pixel resolution.

The pixel resolution is evidently different for all measurements. Since the pixel resolution is governed by the camera's spatial and temporal resolution capabilities, by the region of interest being captured, by the distance between the camera and the tool, and by the features of the tool, and, since we use two different cameras and measure three different tools, each with different features, the pixel resolution is different for all. The pixel resolution being different for all measurements will be useful to check for the robustness of the proposed vision-based motion registration schemes.

A white monotone background was used to minimize the influence of background noise in measurements. For measurements with the high-speed camera, a DC light was used to adequately illuminate the tool. Representative frames showing the field of view for all tools are shown in the insets in Fig. 1(a–c). The smartphone's camera records videos in colour and in the MP4 format. These video recordings lasted ~ 2 s, resulting in less than 100 MB of data per video. Individual frames from these videos were read using the *VideoReader* function in MATLAB – the post processing tool used for all analysis herein. The high-speed camera recordings were ~ 1 s in duration and individual frames were saved in an uncompressed TIFF format, resulting in ~ 8 GB of data for each video. All images were converted to grayscale using the *rgb2gray* function within MATLAB, and these were further processed to extract motion – as is discussed next.

Extraction of tool displacements from video recordings

All grayscale images are 2D matrices of $m \times n$ pixels. Each pixel contains light intensity information corresponding to features of the tool. $I(x, y)$ is this intensity distribution. When the tool vibrates, pixel intensities change, i.e., the intensities become time dependent, becoming $I(x, y, t)$. To extract tool displacements from these changing intensities, this paper discusses the use of three methods.

The first method detailed herein involves the detection of the vibrating edge of each tool in each frame by computing intensity gradients, and then subsequently tracking the same vibrating edge across all frames to obtain a sense of the motion of the tool. Since the edge detection and tracking methods are influenced by image noise, and by the method of evaluating intensity gradients, we describe the commonly used methods, and results with these are presented in the section on 'comparative analysis'.

The second method detailed herein uses the principles of optical flow to estimate displacements in the directions of the intensity gradients only. Displacements are estimated by monitoring pixel intensities between pairs of images, i.e., between $I(t)$ and $I(t + \Delta t)$, and every subsequent pair of images, wherein Δt is the sampling time ($1/\text{fps}$). Since gradient-based optical flow is also sensitive to method of evaluating the gradient, and to image noise, we also describe methods to address these issues, and results with these are also presented in the section on 'comparative analysis'.

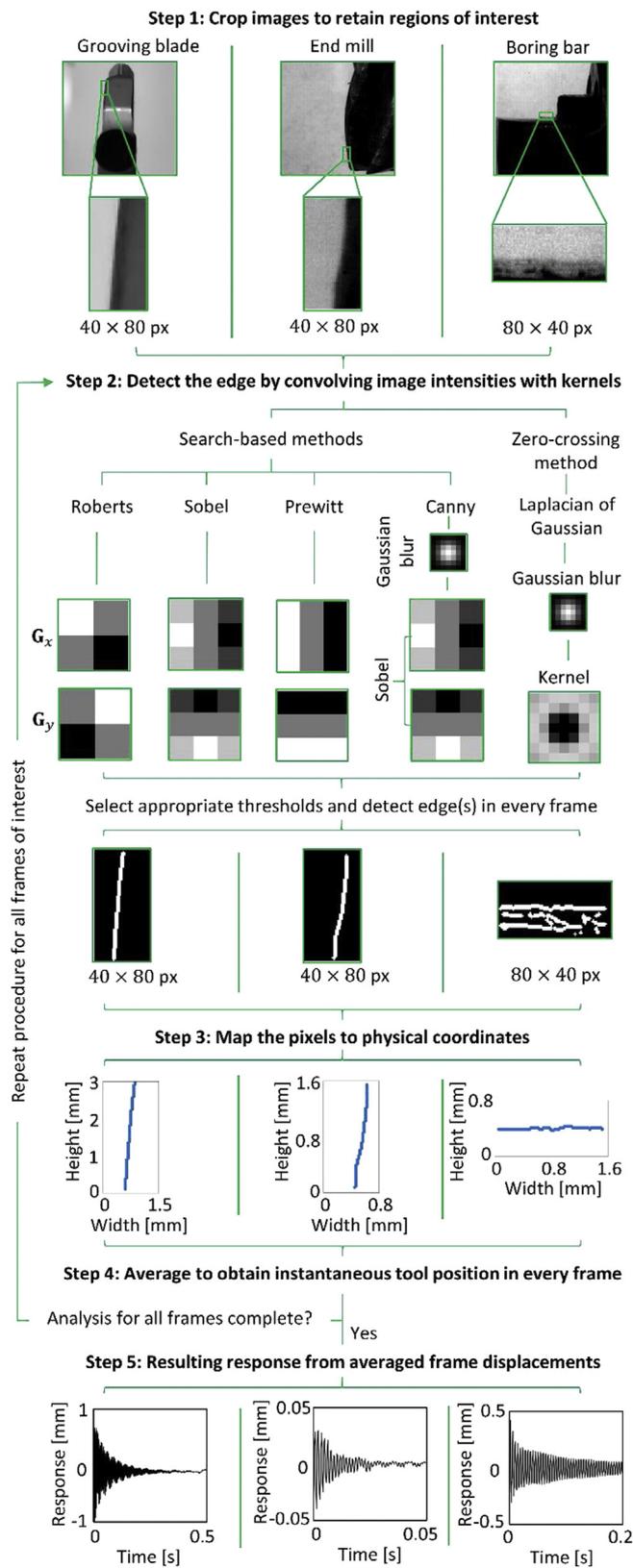


Fig. 2. Overview of the proposed edge detection and tracking schemes to extract tool point displacements.

The third method detailed herein is based on well-established DIC methods. Displacements are estimated by maximizing cross-correlation functions between successive pairs of images while keeping the reference image the same. Since DIC methods are also

sensitive to image noise, and to the size of the subsets within the regions of interest that are to be correlated, we discuss methods to deal with these issues, and results with these too are presented in the section on ‘comparative analysis’.

Edge detection and tracking

An overview of the procedure to detect and track edges for the three tools of interest is illustrated in Fig. 2. There are five main steps. The first step involves cropping all images to retain regions at or near the cutting edge(s) of the tool. Regions of interest spans 40 × 80 pixels (or 80 × 40) for all the measurements herein. The second step involves detecting edges in each frame. The third step involves mapping the detected edge in every frame to its respective in-plane coordinates. Since it is of interest to estimate motion only in the direction of excitation, the fourth step involves averaging the mapped values to result in the relative instantaneous position of the tool for every frame in the direction of interest. The fifth and final step involves repeating steps two to four for all frames of interest. Detailed edge detection procedures are outlined next. For the boring bar, Fig. 2 shows images for the representative case of using the high-speed camera.

The second main step involves detecting the edge in each frame for each tool by finding regions within each image where the image brightness (intensities) change sharply. To find these regions, this paper explores the efficacy of the commonly used search-based methods [27–30] and of the ‘zero-crossing’ method [31]. The search-based methods detect edges by finding the intensity gradient of the image by computing the first order derivatives of the intensity function. The performance of these search-based methods is governed by the choice of the gradient kernel to be convolved with the image intensity that helps find the edge by finding the right gradient direction. The zero-crossing method on the other hand involves evaluating the second order derivative of the image intensity to find regions of zero-crossings that are deemed to be the edges of interest.

In general, for the search-based methods, if the intensity of the cropped frame is I , when this is convolved with gradient kernels G_x and G_y in the X and Y directions, i.e. when $I_x = I * G_x$ and $I_y = I * G_y$, the gradient magnitude and direction become:

$$|H| = \sqrt{I_x^2 + I_y^2}, \quad \Theta = \arctan(I_y/I_x), \tag{1}$$

wherein I_x and I_y are the gradients in the X and in the Y directions, respectively, and H and Θ are the gradient magnitude (edge strength) and direction (edge direction), respectively. Eq. (1) is a generalized representation for the projection coordinates being x and y , and is applicable in a slightly modified form to all three tools of interest to account for the camera’s projection coordinates being different for the three tools.

For the search-based methods, the gradients are computed using the kernels proposed in Ref. [27–29]. Following the method proposed by Roberts [27], the kernel would be:

$$G_x^R = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad G_y^R = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{2}$$

The kernels in Eq. (2) are diagonals oriented at 45° and 135°, respectively. Since these kernels are thought to be more suited to detect diagonal edges in frames [27], we also explore using the Sobel kernels suggested in Ref. [28], which are of the form:

$$G_x^S = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad G_y^S = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{3}$$

As is evident from Eq. (3), the kernels look like horizontal and vertical edges and might be more suited to the application of

interest herein, in which tools usually have well-defined sharp orthogonal features. Since a similar kernel as in Eq. (3), was also proposed by Prewitt [29], we also find gradients using those kernels, which take the form of:

$$G_x^P = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}, \quad G_y^P = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}. \quad (4)$$

Though the Prewitt kernels are similar in structure to the Sobel operator in Eq. (3), since the Prewitt operator is made up of only ones, it does not place any emphasis on the pixels that are closer to the middle of the frame [29].

On estimating the gradient magnitude using any/all the above kernel types, we apply thresholds to decide if edges are present or not at an image point. In general, lower thresholds detect more edges, and are susceptible to noise, whereas higher thresholds may miss subtle edges. Since the choice of appropriate thresholds is non-trivial, we also explore the use of multiple thresholds to find edges – as was suggested by Canny [30]. Furthermore, the use of the Canny edge detector also helps address issues with image noise that was not addressed with the use of the Roberts, Sobel, and/or Prewitt operators.

The Canny edge detector [30] uses the Sobel kernel (Eq. (3)) and involves pre- and post-processing steps to deal with image noise, and to find appropriate thresholds, respectively. Before the image is operated on with the Sobel kernel, the image is convolved with a Gaussian kernel, K:

$$K(x, y) = \frac{1}{(2\pi\sigma^2)} \exp\left(-\frac{(x-d-1)^2 + (y-d-1)^2}{(2\sigma^2)}\right), \quad 1 \leq x, y \leq (2d+1) \quad (5)$$

wherein σ is the standard deviation and $(2d+1) \times (2d+1)$ is the size of the Gaussian kernel. Established practice suggests the size of the Gaussian kernel to be three times the standard deviation in every direction. And, since we have a 2D matrix, the suggested size is 6σ , rounded to an odd integer value. For example, for the case of $\sigma = 3$, the size of the kernel will become 19×19 pixels, making $d = 9$. The choice of σ in turn depends on the suspected level of noise in the image. Since the Gaussian kernel acts as a low pass filter, it suppresses higher frequency detail, i.e., noise, and edges. Since the filter blurs everything smaller than itself, the influence of its size on detection of edges in the regions of interest is necessary and is systematically characterized herein.

On convolving the image with the Gaussian kernel, it is further convolved with the Sobel kernel to get image gradients using Eq. (1). This is followed by the post-processing step of finding the ‘largest’ edge using non-maximum suppression [30] which helps partly resolve the presence of thick and thin edges. This involves retaining only those intensity values in $|H|$ which are large compared to their immediate neighbours in the direction of Θ . And, finally, the issue of differential brightness in the detected edge(s) is addressed by using a double threshold. Intensity values below the lower threshold are discarded and those with values higher than the upper threshold are retained, and intensity values in between the two thresholds are retained only when at least one immediate neighbour in any direction possesses intensities higher than the upper threshold. Performance of these search-based methods and the sensitivity of the tool’s response to appropriate choice of thresholds(s) and kernel sizes is systematically characterized in the section on ‘comparative analysis’.

Performance of the search-based edge detection method is also contrasted herein with the zero-crossing method, otherwise also

referred to as the Laplacian of Gaussian (LoG) method [31]. The LoG kernel is defined as:

$$G^L(x, y) = \frac{-1}{\pi\sigma^4} \left(1 - \frac{(x^2 + y^2)}{2\sigma^2}\right) \exp\left(-\frac{(x^2 + y^2)}{2\sigma^2}\right) \quad (6)$$

wherein σ is still the standard deviation. The LoG kernel is symmetric. The number of zero crossings are found to identify edges. After the image is convolved with this kernel, thresholding is done differently compared to other search-based methods. In this method, eight neighbours around the pixel of interest are considered. If the difference between the maximum and minimum value in all the nine pixels is greater than the threshold and, if there is a difference in sign between the pixel in the centre and the maximum/minimum value, then that pixel corresponds to an edge and it is retained. Having detailed the main step of edge detection in each frame, procedures for extracting displacements from the detected edge continue as outlined in Fig. 2.

Optical flow-based motion tracking

An overview of the procedure to extract tool displacements from the expanded optical flow-based motion tracking scheme is summarized in Fig. 3. Extraction of tool motion involves four main steps. The first step is the same as in the edge detection and tracking scheme and involves cropping all images to retain regions at or near the cutting edge(s) of the tool. Fig. 3, like Fig. 2, only shows the cropped image of the boring bar for when it was measured using the high-speed camera. The second main step involves estimating displacements by monitoring pixel intensities between pairs of images using the gradient-based optical flow-based schemes outlined in Refs. [6,10]. Since the gradient-based optical flow scheme is sensitive to image noise, we expand the optical flow-based scheme proposed in Ref. [6] by using Gaussian blurs like in the case of the Canny edge detector. We also explore efficacy of different methods to evaluate the gradient. And, finally, since optical flow-based schemes are also sensitive to the regions of interest being evaluated, we also present expanded analysis by using an edge detector in combination with optical flow to evaluate motion of the vibrating tool. The displacements thus obtained from every pair of images are averaged over the range of interest in the third step, and the procedure is repeated for every successive pairs of images to finally extract tool motion in the final step. Details of the main steps are described next.

The method of optical flow assumes that brightness is conserved [25,26], i.e., the light intensity of a point on the tool x_i, y_j that moved by a small amount to another location $x_i + \Delta x, y_j + \Delta y$ in a short time Δt is constant such that:

$$I(x_i, y_j, t) = I(x_i + \Delta x, y_j + \Delta y, t + \Delta t) \quad (7)$$

Assuming small motion, the image intensity function can be approximated by a first order Taylor expansion as [10]:

$$I(x_i + \Delta x, y_j + \Delta y, t + \Delta t) = I(x_i, y_j, t) + \frac{\partial I}{\partial x} \Delta x \Big|_{x=x_i} + \frac{\partial I}{\partial y} \Delta y \Big|_{y=y_j} + \frac{\partial I}{\partial t} \Delta t \Big|_t \quad (8)$$

Eq. (7) and (8) are generalized representations for the projection coordinates being x and y . And, even though the problem could be reduced to a one-dimensional form – since we are only interested in motion along the direction of excitation, the formulations are kept generalized for estimating in-plane motion, as necessary.

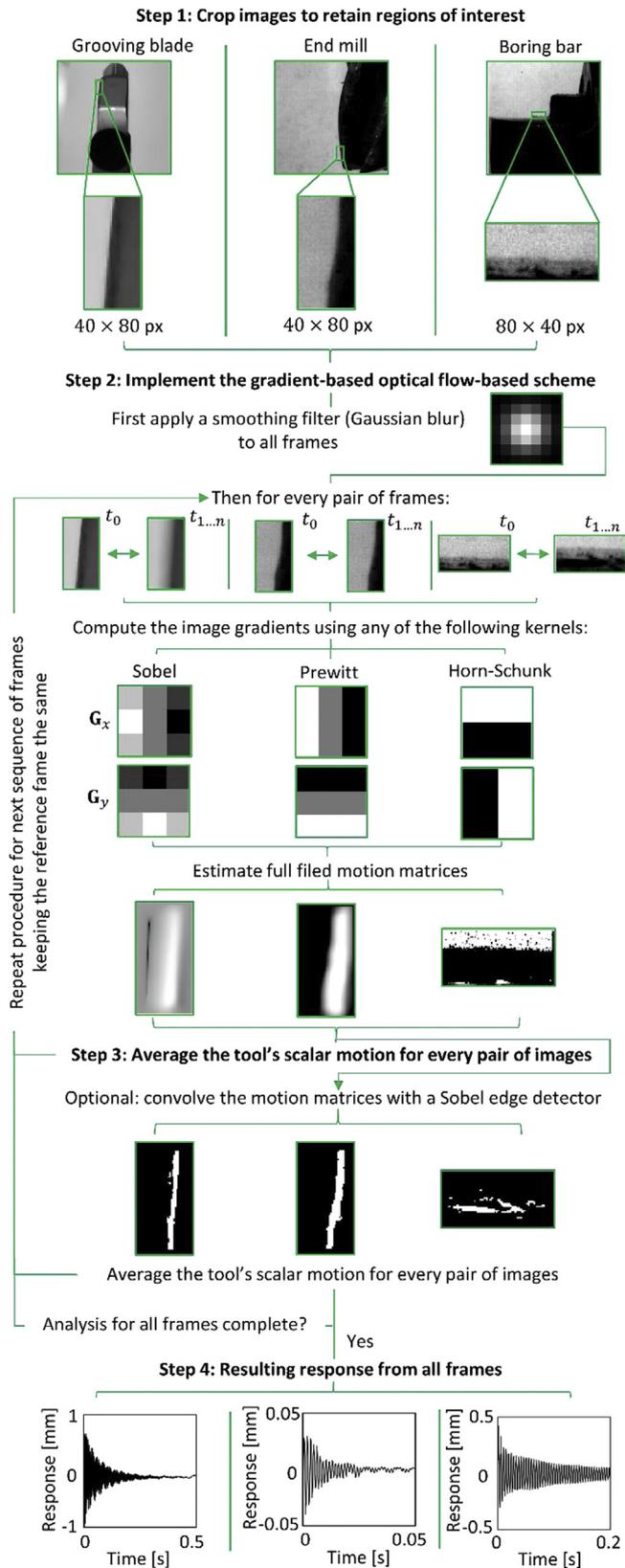


Fig. 3. Overview of the proposed optical flow-based motion tracking scheme to extract tool point displacements.

For small motion and for short time intervals, the optical flow equation from Eq. (7) and Eq. (8) becomes [6,10]:

$$\frac{\partial I}{\partial x} \Delta x \Big|_{x=x_i} + \frac{\partial I}{\partial y} \Delta y \Big|_{y=y_i} + \frac{\partial I}{\partial t} \Delta t \Big|_t = 0. \quad (9)$$

Since $\frac{\partial I}{\partial t} \Delta t$ at some time, t in Eq. (9) is the change in intensity at a point between two frames, i.e., since $\frac{\partial I}{\partial t} \Delta t \Big|_t = I(x_i, y_j, t + \Delta t) - I(x_i, y_j, t)$, Eq. (9) can be rewritten as:

$$\frac{\partial I}{\partial x} \Delta x \Big|_{x=x_i} + \frac{\partial I}{\partial y} \Delta y \Big|_{y=y_i} = I(x_i, y_j, t) - I(x_i, y_j, t + \Delta t). \quad (10)$$

Though intensities ($I(t)$ and $I(t + \Delta t)$) are known, and the gradients $\partial I/\partial x$ and $\partial I/\partial y$ can be determined, Eq. (10) has two unknowns in Δx and in Δy , and to solve for these, usually an additional smoothness constraint is introduced [25]. However, since displacements are of interest only in the direction of the intensity gradients, Eq. (10) can be rewritten as [6,10]:

$$|\nabla I| \Delta M(x_i, y_j, t) = I(x_i, y_j, t) - I(x_i, y_j, t + \Delta t), \quad (11)$$

wherein $|\nabla I| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 \Big|_{x=x_i} + \left(\frac{\partial I}{\partial y}\right)^2 \Big|_{y=y_i}}$ is the intensity gradient,

and ΔM is the displacement in the direction of the intensity gradient. Since tool displacements are assumed to be small, the displacements are evaluated by monitoring pixel intensities between the reference image $I_0(x_i, y_j)$ and subsequent images

$I(x_i, y_j, t)$:

$$M(x_i, y_j, t) = \frac{I_0(x_i, y_j) - I(x_i, y_j, t)}{|\nabla I_0|} \quad (12)$$

wherein $|\nabla I_0|$ is the intensity gradient matrix computed with respect to the reference image. This gradient can be evaluated by convolving the image intensity with any of the Roberts, Sobel, and/or Prewitt kernels described in the subsection discussing the edge detection schemes, or alternatively, the default kernel suggested in Ref. [25] may also be used, in which case the kernel would be of the form of:

$$\mathbf{G}_x^{\text{HS}} = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{G}_y^{\text{HS}} = \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}. \quad (13)$$

Sensitivity of the performance of the optical flow method to the choice of the kernel used to evaluate gradients is characterized in the section on 'comparative analysis'. The image derivatives required to evaluate intensity gradient in Eq. (12) is given by Eq. (14). Since the size of kernels are different, the gradients are normalized with respect to the kernel size such that the magnitude of the mean pixel intensity remains the same in the resultant image gradient matrix:

$$I_x|_t = \frac{1}{(p-1) \cdot \text{sum}(|\mathbf{G}_x|)} [(I(t) * \mathbf{G}_x + I_0 * \mathbf{G}_x),$$

$$I_y|_t = \frac{1}{(p-1) \cdot \text{sum}(|\mathbf{G}_y|)} [(I(t) * \mathbf{G}_y + I_0 * \mathbf{G}_y). \quad (14)$$

wherein $\text{sum}(|\mathbf{G}_x|)$ and $\text{sum}(|\mathbf{G}_y|)$ represents the sum of the absolute entries, and 'p' represents the size of the kernel matrix.

Since the resulting motion matrix, \mathbf{M} contains motion estimated for each pixel in the frame, i.e., since it is a full-field estimate, in the third step, we extract the tool's scalar motion from the full-field as:

$$x(t) = \mathbf{M}(t) * \frac{1}{mn} \text{ones}(m, n) \quad (15)$$

wherein $\text{ones}(m, n)$ represents a matrix of size $m \times n$ with all input entries being unity.

Despite the use of the Gaussian blur, since the full-field contains the tool as well as some background, motion evaluated using Eq. (15) may be influenced by background noise. To avoid the pitfalls of displacement being corrupted by noise, we convolve the motion matrix with a Sobel edge detector, to result in the tool's averaged displacement over the detected tool's edge as:

$$x(t) = \mathbf{M}(t) * \left[\frac{1}{\text{nnz}(\text{edge}(\mathbf{I}))} \text{edge}(\mathbf{I}) \right]_t \quad (16)$$

wherein the $\text{edge}(\cdot)$ operator represents the Sobel operator (see Eq. (3)) and $\text{nnz}(\cdot)$ represents number of non-zero entries in an input matrix. The Sobel edge detector used within Eq. (16) involves an implementation of the Sobel edge detection scheme using a threshold limit to result in the averaged tool displacement between the frame of interest and the reference frame. Comparative evaluation of both methods of extracting the tool's displacement, i.e., the standard method using Eq. (15) and the expanded method using Eq. (16) is presented in the section on 'comparative analysis'. Above procedures are repeated in the final step by selecting the next frame in sequence as shown in Fig. 3 – to finally result in the tool's motion over all frames of interest.

Digital image correlation

An overview of the procedure to extract tool displacements for all three tools of interest using DIC is shown in Fig. 4. For the boring bar, Fig. 4 only shows the representative cropped image acquired using the high-speed camera. The DIC procedure comprises of three main steps. The first step is the same as discussed above and involves cropping all images to retain regions at or near the cutting edge(s) of the tool. In the second and the main step, we extract displacements by spatial correlation of subsets within regions of interest on a frame of interest with that of a reference image. This procedure is repeated in the third step for every subsequent frame keeping the reference frame the same. Prior to estimating motion in the second and main step, all images are convolved with a Gaussian blur to minimize the influence of image noise, as necessary.

To estimate motion in the second step, we find the maximum of the correlation array between pixel intensity array subsets between a reference image and the current image of interest. Since we have no speckle pattern to choose a subset from, we choose the region around the edge(s) of the tool of interest as the subset in the reference image. These regions have a naturally occurring contrast. These subsets in the reference images, i.e., at t_0 are highlighted in Fig. 4. Since we are interested in finding the rigid translation of the tool from one frame to the next, which is obtained by correlating the subsets between the frame of interest and the reference frame, the subset in the frame of interest is traversed incrementally (in steps of the size of one pixel) across the width/height of the image to find the maximum correlation between each pair of subsets for every pair of images. Two such subsets in two non-reference images are shown schematically in Fig. 4. To find the correlation between each subset pair for every pair of images, we evaluate the zero normalized cross-correlation

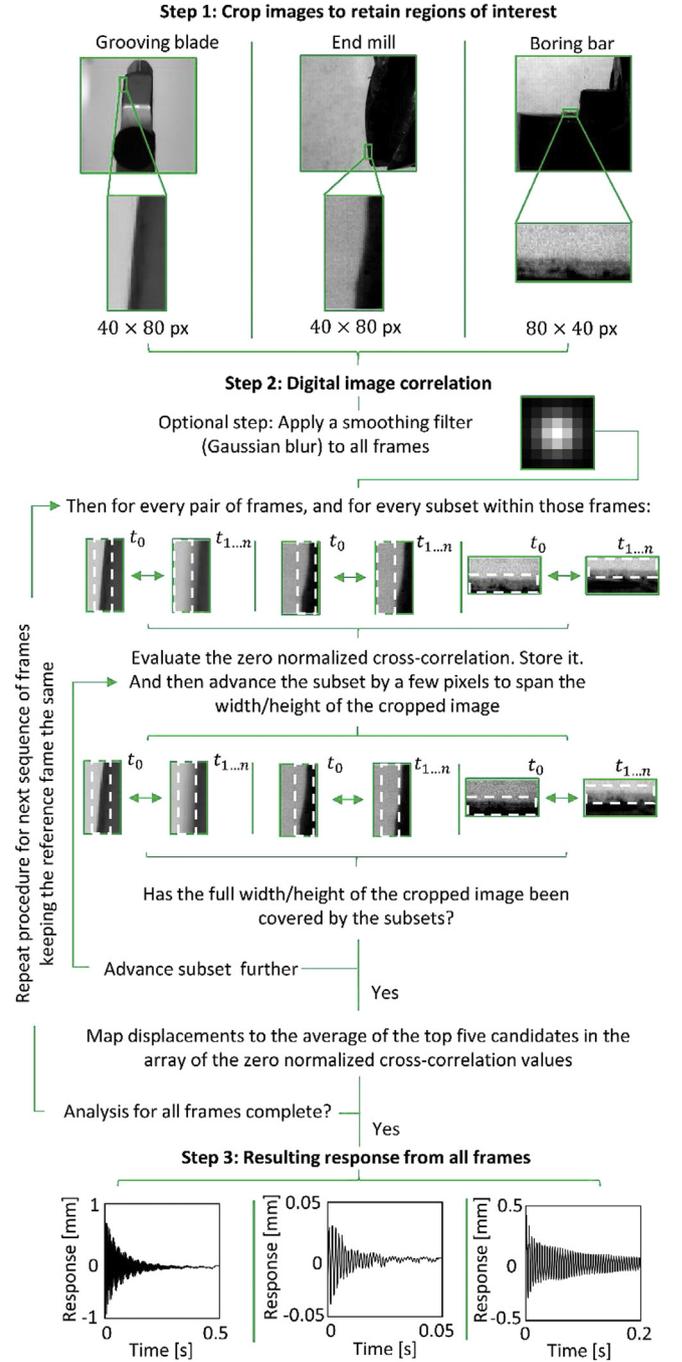


Fig. 4. Overview of the digital image correlation scheme to extract tool motion.

function defined as [23]:

$$\gamma(u, v) = \frac{\sum_{x,y} [f(x, y) - \bar{f}_{u,v}] [g(x - u, y - v) - \bar{g}]}{\left\{ \sum_{x,y} [f(x, y) - \bar{f}_{u,v}]^2 \sum_{x,y} [g(x - u, y - v) - \bar{g}]^2 \right\}^{0.5}} \quad (17)$$

wherein f is the pixel intensity at x, y and the sum is over x, y under the window containing the subset in the reference image. Similarly, g is positioned at u, v , and is also the image intensity summed over x, y under the window containing the subset in the image of interest. \bar{f} and \bar{g} are the mean values of the intensity matrices f and g respectively.

Though the cross-correlation function in Eq. (17) can be maximized to extract displacements using one of many

optimization schemes [10,22–24], we prefer a simpler implementation that is based on storing $\gamma(u, v)$ for every pair of subsets for every pair of images under consideration. From this stored array, to find translation of the tool between the pair of images under consideration, we find and average the locations corresponding to the five highest cross-correlation functions, and from that averaged location, we subtract the location of the subset in the reference image. This procedure is easy to automate, and is not very sensitive to the initial guess, or to the method of optimization, and is hence preferred herein.

Having estimated the displacement of the tool between a pair of images, the above outlined procedure is repeated as shown in Fig. 4 for every subsequent frame of interest while always keeping the

reference frame the same. Sensitivity of the estimated response to the size of the subset, and/or to the size of the Gaussian blur is also checked, and those results are discussed in the next main section. Though we are interested in estimating motion only in the direction of excitation, procedures outlined are valid for in-plane motion estimation. The procedure may be further refined using subpixel level registrations [6,10].

Comparative analysis of tool response

This section compares tool response estimated using the three methods outlined above. All results for all three tools are benchmarked against twice integrated accelerations. Since the

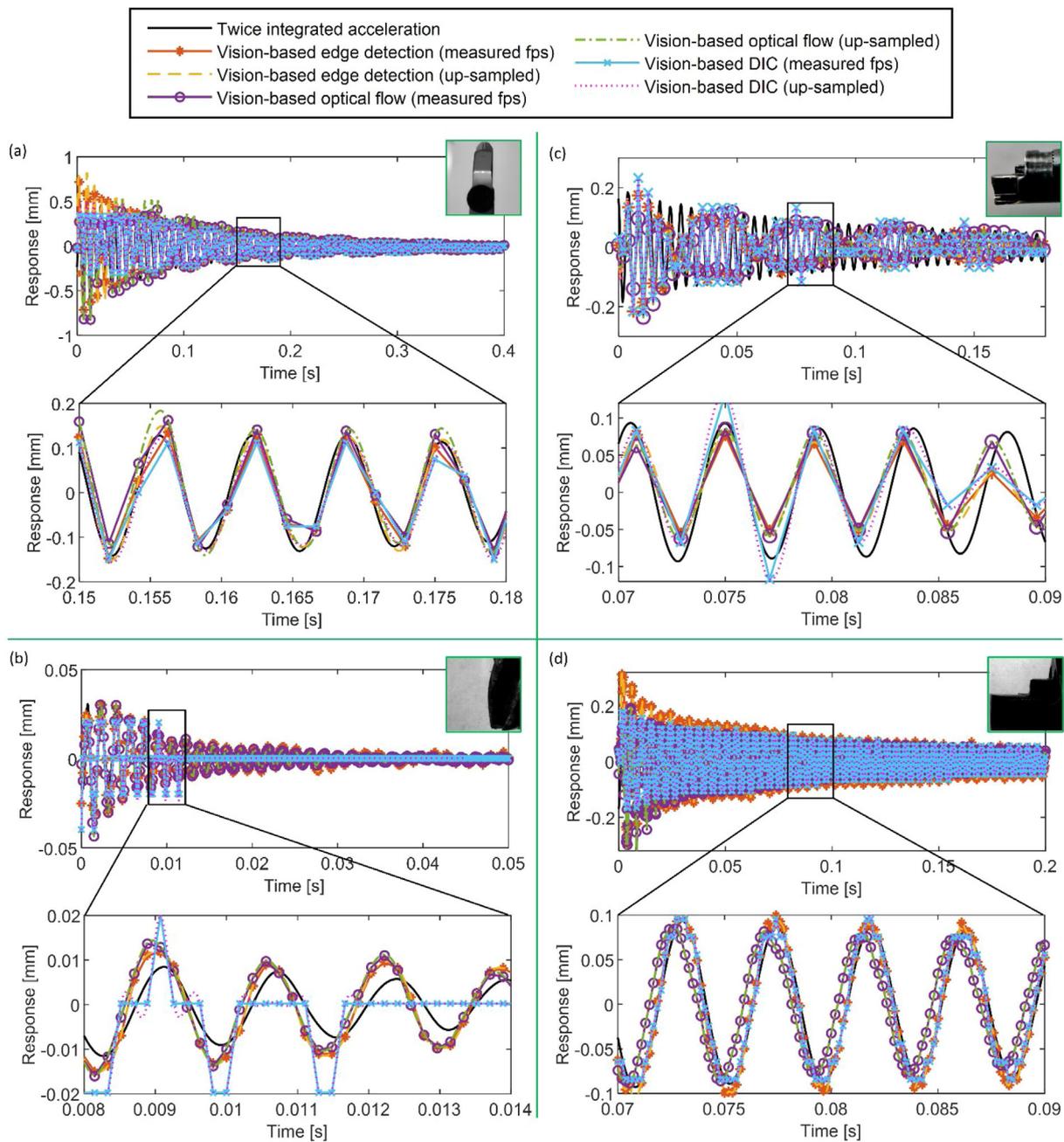


Fig. 5. Comparison of the twice integrated measured acceleration with vision-based motion estimated using the regularly sampled video and with extracted motion being up-sampled. Vision-based motion estimates include those with the edge detection and tracking scheme, the optical flow-based motion tracking scheme, and the DIC scheme. (a) Response for the grooving blade with video recorded at 480 fps, (b) response for end mill with video recorded at 5400 fps, (c) response for the boring bar with video recorded at 480 fps, and (d) response for the boring bar with video recorded at 5400 fps.

acceleration data was sampled at the rate of 25.6 kHz, and since the smartphone camera-based and the high-speed camera-based measurements are sampled at rates of 480 fps and 5400 fps, respectively, to make comparisons meaningful, all vision-based motion measurements were up-sampled to 25.6 kHz using a *spline* interpolation scheme, as was also done elsewhere in [39].

Comparative analysis of the twice integrated measured acceleration response with vision-based motion estimated using regularly sampled video and with extracted motion being up-sampled is shown in Fig. 5. For the edge detection and tracking scheme, results in Fig. 5 are limited to the use of the Canny edge detector with optimal thresholds and size of the Gaussian blur.

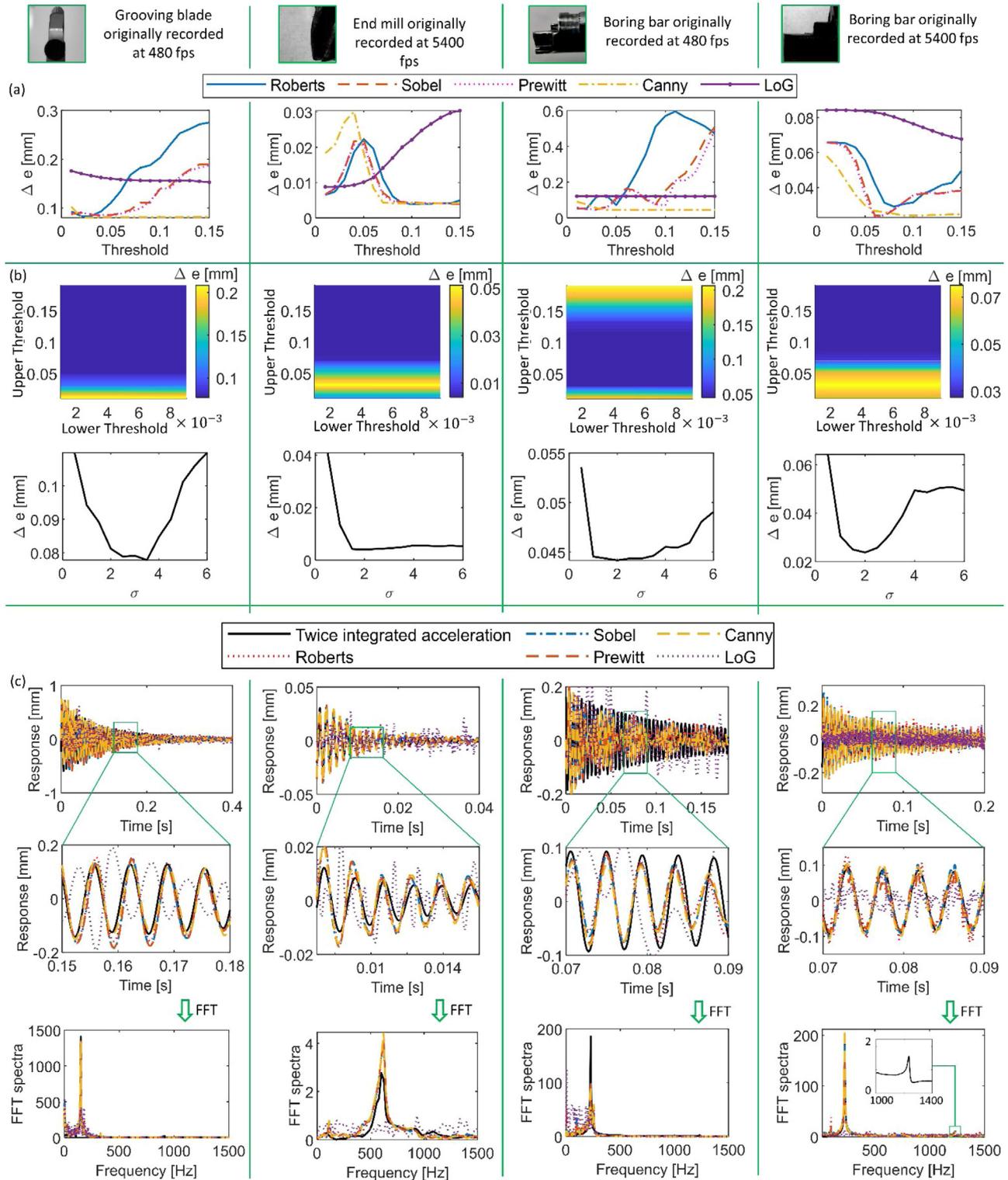


Fig. 6. (a) Influence of kernel type and thresholds – characterized by the RMS error ($\Delta\epsilon$); (b) Influence of changing upper and lower thresholds characterized by the RMS error ($\Delta\epsilon$) – shown in the colour bar along with influence of size of the Gaussian blur for the case of the Canny edge detector; (c) Comparison of the tool's displacement and its frequency spectra with displacements extracted using different edge detection and tracking schemes.

Response being influenced by the kernel type, by thresholds, and by the size of the Gaussian blur is separately characterized in the subsection discussing results with the edge detection schemes. Similarly, results with the optical flow-based motion tracking scheme shown in Fig. 5 are only for the case of extracting displacements by convolving the motion matrices evaluated using a Sobel operator with a Sobel edge detector. And response being influenced by the kernel type and by the size of the Gaussian blur, as well as by the method of extracting motion from the full field is separately discussed in the subsection discussing the optical flow-based schemes. For the DIC results shown in Fig. 5, results are shown with the optimal size of the subset and of the Gaussian blur. And response being influenced by different subset sizes and/or by different sizes of the Gaussian blur is separately addressed in the subsection discussing results with the DIC scheme.

As is amply evident from Fig. 5(a) which shows response for the grooving blade with video recorded at 480 fps, and from Fig. 5(b) which shows response for the end mill with video recorded at 5400 fps, and further from Fig. 5(c–d) which shows response for the boring bar with videos sampled at 480 fps and at 5400 fps, respectively, though the trend of the motion extracted using the vision-based approaches reasonably approximates the twice integrated measured accelerations, the up-sampled vision-based response in all cases is in closer agreement with the twice integrated measured accelerations. Also evident from Fig. 5 is that for all cases and all tools, the response estimated using all three vision-based methods are in reasonably good agreement with each other. It is further evident from Fig. 5 that the grooving blade and the boring bar vibrate with amplitudes larger than the end mill, and that the three tools have a noticeably different period of vibration.

For the case of the end mill, results for which are shown in Fig. 5(b), motion estimated with the DIC scheme appears to be inconsistent with twice integrated accelerations at time instants of when the tool response has decayed for it to have a magnitude less than the pixel resolution. DIC schemes are generally recommended to be used with objects with speckle patterns on them [7,9,10], as speckle patterns help track small motion even when the motion is less than or of the order of the pixel size. Since we have no speckle pattern to choose a subset from, and since we choose the region around the edge(s) of the tool of interest as the subset in the reference image, the estimated tool motion being inconsistent with the twice integrated accelerations is not surprising. To register motion smaller than the pixel size, subpixel motion registration algorithms [6,10] that expand the region of interest in the cropped frame using advanced interpolation schemes may prove useful, but are beyond the scope of our present work and can be considered in future follow on research. Since the response amplitudes of the boring bar and the grooving blade are larger than the pixel resolutions, motion estimated using the DIC scheme works well in those cases – evident in Fig. 5(a, c, and d).

For the case of the boring bar's response being estimated with the video being recorded at 480 fps – see Fig. 5(c), there appears to be a beating like phenomenon taking place. The fundamental mode of the boring bar occurs at ~ 226.5 Hz – as is estimated from accelerometer data sampled at 25.6 kHz. Since video was recorded at 480 fps, i.e., the sampling frequency (f_s) was 480 Hz, and since the signal frequency (f_0) at ~ 226.5 Hz lies just below the Nyquist frequency ($f_N = f_s/2$) of 240 Hz, the occurrence of a low frequency behavior beating phenomenon is not entirely unusual even though the signal does not, in fact, contain this frequency [40]. This distortion of the signal occurs even when the Shannon sampling theorem is strictly satisfied. The beat frequency, f_{beat} , from Fig. 5(c) can be estimated to be ~ 26 Hz, and is very near the theoretically

expected $f_{\text{beat}} = 2(f_N - f_0)$ of 27 Hz. Since this is a pseudo frequency, and is an artefact of inappropriate sampling, it is not a real mode, and does not influence modal parameter identification discussed subsequently.

For the boring bar's motion estimated using higher frame-rate recordings of 5400 fps, no such beating phenomenon is observed – see Fig. 5(d). Nor is any beating phenomenon observed for the end mill vibrating around ~ 612 Hz and being recorded at the rate of 5400 fps – see Fig. 5(b). Similarly, even for the case of the grooving blade recorded with the lower frame rate camera with 480 fps, since the dominant mode occurs around ~ 151 Hz, beating is also not observed in that case – see Fig. 5(a).

Seeing that the up-sampled vision-based response is smoother and has the same size as that of the twice integrated measured accelerations, all analysis hereafter in the paper is made using the up-sampled vision-based response. To aid comparisons, we define a root mean square (RMS) error metric as:

$$\Delta\epsilon = \frac{1}{r} \sqrt{\sum_{i=1}^r (x_i^a - x_i^v)^2} \quad (18)$$

wherein \mathbf{x}^a and \mathbf{x}^v are displacement vectors obtained using the twice integrated measured accelerations and the up-sampled vision-based response, respectively, and 'r' is the total number of discrete points in the displacement vector.

Response obtained with edge detection and tracking schemes

A summary of the performance of all edge detection and tracking methods with changing kernels, thresholds, and Gaussian kernels (where applicable) for all tools is provided in Fig. 6. Since the kernels proposed by Roberts (see Eq. (2)), by Sobel (see Eq. (3)), by Prewitt (see Eq. (4)), and in the zero-crossing LoG method (see Eq. (6)) are all different, how the choice of kernel influences the extracted response is characterized separately for all tools. Since thresholding also influences edge detection, the RMS error ($\Delta\epsilon$) of the displacements estimated with different kernel types with changing levels of thresholds is shown in Fig. 6(a). The Roberts, Sobel, Prewitt, and LoG methods of edge detection each work by defining only a single threshold value above which edges are retained. The Canny edge detector on the other hand works with two thresholds, an upper and a lower within which edges are retained. Hence, for the analysis in Fig. 6(a), the lower threshold in the Canny method is kept constant at 0.005, and only the upper threshold is varied. Influence of changing upper and lower thresholds with the Canny edge detector are separately characterized in Fig. 6(b) for all tools of interest. Furthermore, since the performance of the Canny edge detector and of the LoG method is also governed by the size of the Gaussian blur (see Eq. (5) and Eq. (6)), for comparisons in Fig. 6(a), the standard deviation is kept fixed at 2. For the case of the Canny edge detector, influence of the size of the Gaussian blur (by varying the standard deviation) is separately characterized in Fig. 6(b) for all tools. Finally, for thresholds and the size of the Gaussian blurs that result in lower RMS errors, the response for all tools is compared with the twice integrated acceleration in Fig. 6(c).

Fig. 6(a) compares the RMS errors changing with thresholds for the different edge detection schemes. In general, though a low threshold may result in a spurious edge being detected, and a high threshold may miss the real edge, since all tools were measured with different acquisition parameters and under different illuminations, and since each tool has different features, and since these factors interact with each other in complex and occasionally nonintuitive ways, the threshold values at which the RMS error is minimum is different for all tools, and this threshold is also different for different edge detection schemes.

Even though RMS errors are different for all tools, the trends in errors appear to be consistent for the case of recordings made at different frame rates. For example, for the case of the grooving blade and the boring bar recorded at 480 fps, the RMS errors increase with an increase in the threshold value for the case of the Roberts', Sobel, and Prewitt kernels – which is consistent with expectations since higher thresholds result in potentially missed edges. Though low frame rate recordings suggest that RMS errors increase with thresholds, contrary behavior is observed for high frame rate recordings for the case of the end mill and the boring bar recorded at 5400 fps. This contrary and counterintuitive behavior is likely due to the complex interactions between high frame-rate images being darker, the use of DC light sources to illuminate these tools, and due to high frame-rate images resulting in less motion blur but potentially more noise. For the tools recorded with higher frame

rates, there appears to be a critical threshold value beyond which the RMS errors tend to decrease for the case of the Roberts', Sobel, and Prewitt kernels. For the motion of the boring bar extracted with these three schemes, the error being high for lower thresholds can be explained by how lower thresholds cause spurious edges to be detected. However, contrary behavior is observed for the case of the end mill, in which the error is low for lower thresholds.

For the case of the LoG method, for all tools of interest, whether sampled at the low or the high frame-rates, the RMS errors are consistently high, and this is likely due to the LoG method being more suited to detect blobs [31] than horizontal and/or vertical edges – like we see in the case of the tools of interest. The Canny edge detector which uses the Sobel kernel results in the lowest RMS errors for all tools and appears to perform consistently better than the other schemes and this is likely due to its additional

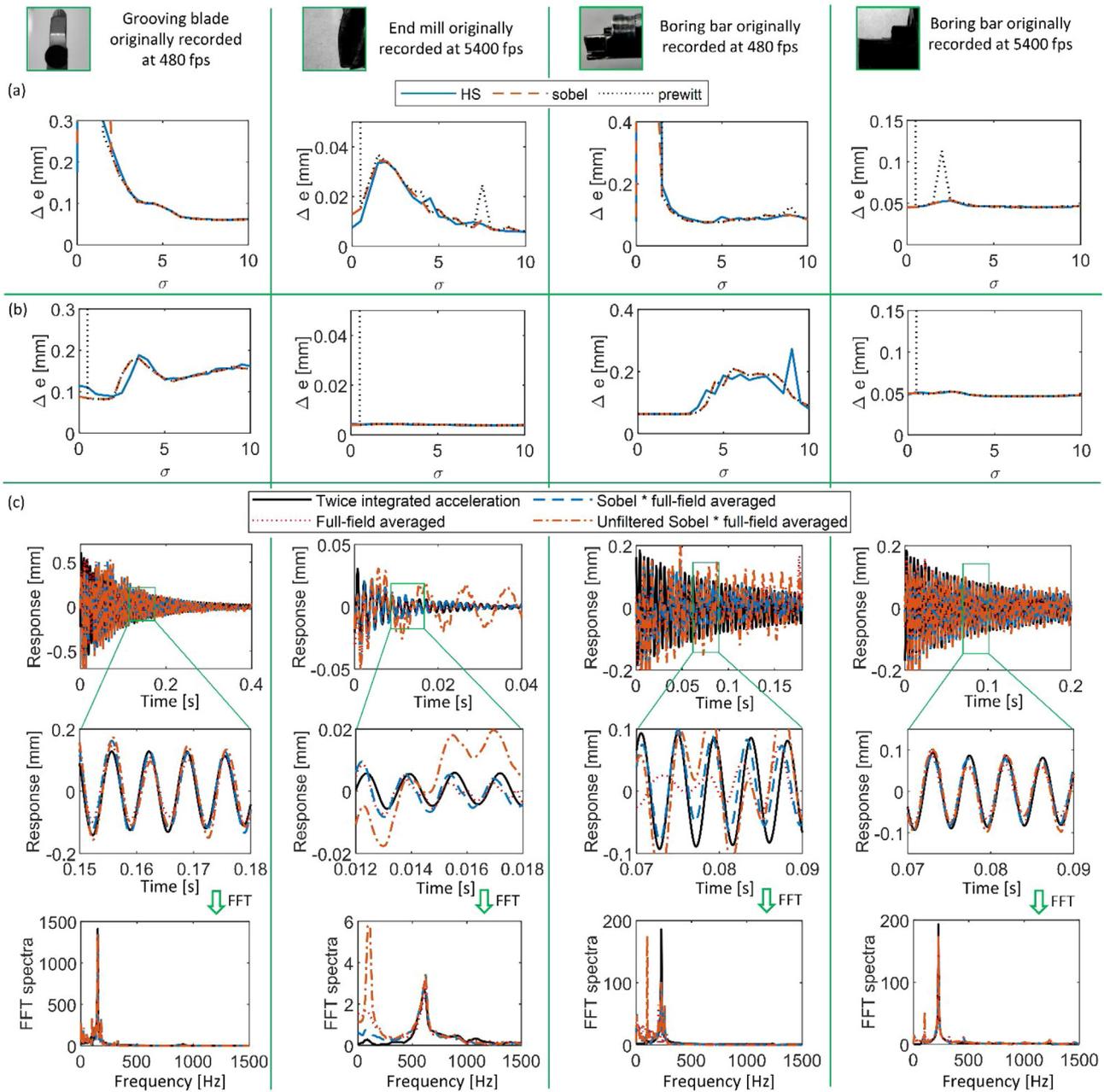


Fig. 7. (a) Comparisons for motion averaged from the full field changing with kernel type and size of the Gaussian blur – characterized by the RMS error (Δe); (b) Comparisons for displacements obtained by convolving the motion matrices with a Sobel operator for changing kernel types and size of the Gaussian blur – characterized by the RMS error (Δe). (c) Comparison of the tool's displacement and its frequency spectra with displacements extracted using different methods.

pre- and post-processing steps to deal with image noise and its use of double thresholds for edge detection.

Results with the Canny edge detector shown in Fig. 6(b) suggest that the RMS error (shown by the colour bar) for all tools appears to be independent of changing the lower threshold value. Results also appear to improve (lower Δe) for increasing the levels of upper thresholds. Improvements appear to saturate for upper thresholds greater than 0.05 except for the case of boring bar recorded at 480 fps where the RMS error again increased with an increase in upper threshold beyond 0.15. Fixing the lower threshold to be 0.005, and the upper threshold to be 0.1, results with changing the size of the Gaussian blur show that the size of the Gaussian blur that results in the lowest RMS error is different for all tools. In general, the effect of Gaussian smoothing is to blur an image, i.e., to attenuate high frequencies more than low frequencies, which in effect, reduces noise in the image. Larger filters, i.e., those with a higher σ will result in greater amounts of noise reduction, with the trade-off however of blurring the image, i.e., farther pixels get to contribute to the new value of the center pixel as opposed to a smaller kernel. Since edge detection is more difficult in blurry images, the RMS errors that are observed to increase for increasing sizes of σ , as

observed in Fig. 6(b) is unsurprising. Overall, for Canny edge detection, an upper threshold of ~ 0.1 and a standard deviation of 2 or 3 appears to perform satisfactorily for all the four experiments discussed in Fig. 6.

Response comparisons shown in Fig. 6(c) suggest that motion estimated using all methods of edge detection, except for the LoG method, are comparable to the twice integrated measured accelerations for all tools. For the LoG method, the motion estimated appears incorrect and noisy – which is confirmed by the frequency content of the response signals. The spectra of the response signals suggests that the grooving blade has a dominant mode at ~ 151 Hz, the end mill has a dominant mode at ~ 612 Hz, and the boring bar has two modes, one at ~ 226.5 Hz, and another at ~ 1235 Hz. Also evident from the spectra is that smartphone-based camera recording video at 480 fps cannot capture the second mode of the boring bar. Further evident from all spectra is that though the peak frequencies are almost the same for all methods of edge detection and tracking schemes for all tools, the magnitudes of the spectra at these peak frequencies is slightly different, and these may influence modal parameter extraction discussed in the subsequent main section.

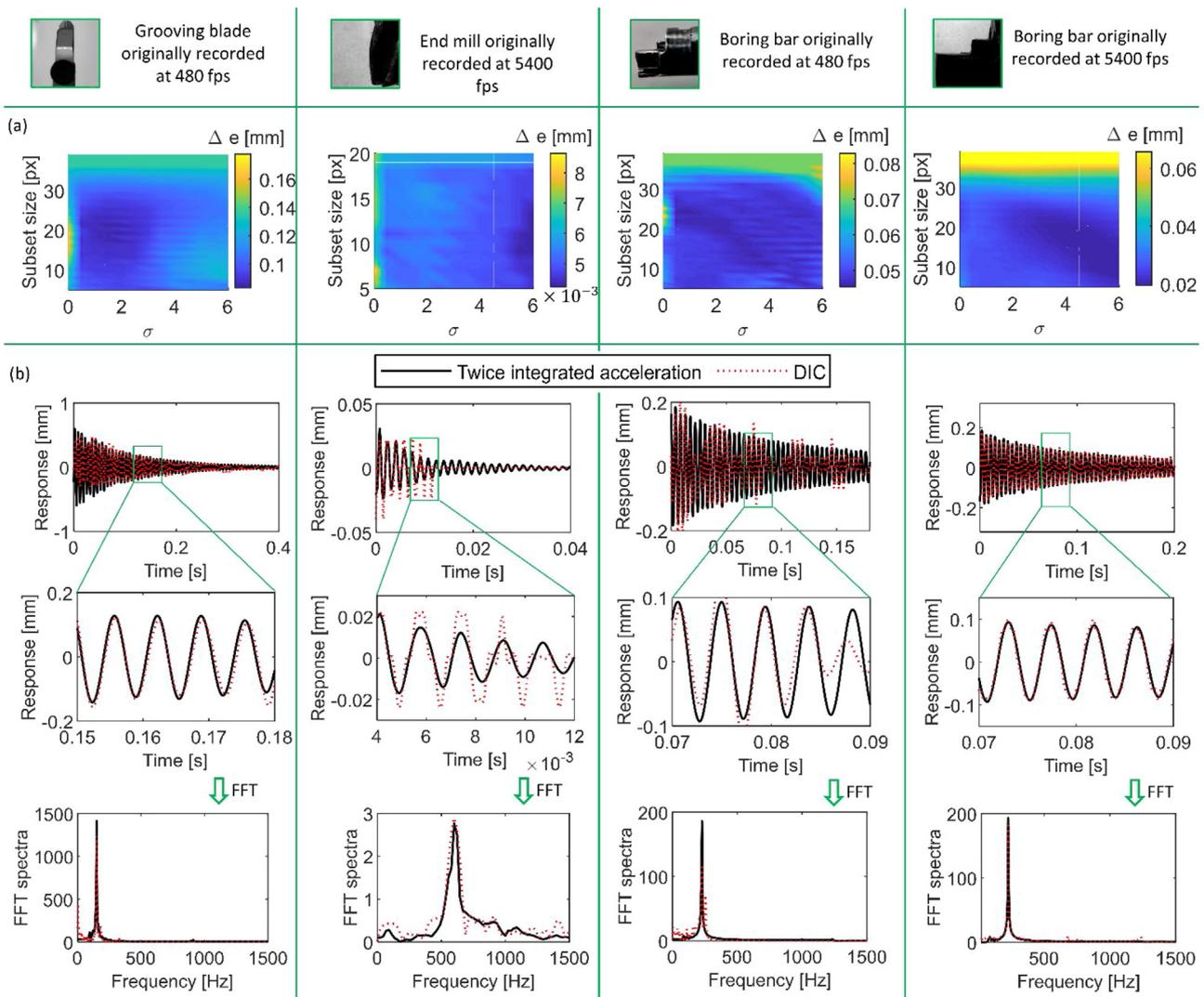


Fig. 8. (a) RMS error (Δe) in the estimated displacements changing with size of subsets and Gaussian blurs; (b) Tool displacement and its frequency spectra for motion estimated using DIC being compared with twice integrated accelerations.

Response obtained using optical flow

Comparative analysis of response extracted with optical flow-based schemes governed by the size of the Gaussian blur, the choice of the kernel used to compute image intensity gradients, and by the method of averaging motion from the full field is summarized in Fig. 7. Comparisons in Fig. 7(a) are characterized by RMS errors for motion averaged from the full field using different kernels and for results changing with the size of the Gaussian blur. Similarly, comparisons in Fig. 7(b) show the displacements obtained by convolving the motion matrices with a Sobel edge detector, when different kernels were used to evaluate the image intensity gradients and for results changing with the size of the Gaussian blur. Finally, results in Fig. 7(c) show response obtained with the size of the Gaussian blur, and kernel choice that results in lower RMS errors, and compares that response with twice integrated accelerations for all tools of interest.

For all results in Fig. 7 that are obtained by convolving the motion matrices with a Sobel operator, the threshold for the operator was fixed at 0.1. This threshold was decided based on the analysis presented in Fig. 6 – even though that analysis was strictly valid only for the edge detection scheme. And even though this 0.1 threshold does not always result in the lowest RMS error for all tools (see Fig. 6(a)), the RMS errors remain relatively low for all tools for this threshold.

Motion estimates using optical flow techniques are particularly susceptible to being corrupted by noise due to flicker because of the assumption of brightness constancy during motion registration. Since all our experiments are carried out in environments illuminated by fluorescent lamps which use core-coil magnetic ballasts to regulate the alternating current through the lamp, and since luminous modulation of fluorescent lamps flickering at twice the frequency of the electrical supply is well-known [41], to eliminate this noise due to flicker, we apply a notch filter at 100 Hz, which is twice the supply frequency of 50 Hz. And even though some measurements are conducted with tools illuminated by DC lights, the ambient light still corrupts the measurement with its flicker and is accordingly filtered. All comparisons in Fig. 7(a–b) are made with the notch-filtered response. As a representative case, Fig. 7(c) also shows the unfiltered response for all three tools that was obtained by convolving the motion matrices with a Sobel edge detector.

From Fig. 7(a–b), besides the odd outlier, it is evident that the difference between RMS errors for the three gradient kernels is small. From Fig. 7(a) it is also evident that, in general, the RMS errors decrease with an increase in size of the smoothing filter. These observations are different than those in Fig. 6(b). Since optical flow-based schemes estimate motion by averaging pixel intensities between the current frame and a reference frame (see Eq. (12) and Eq. (15)), and since the role of the filter in this case is to smoothen both frames, a higher σ will naturally result in greater

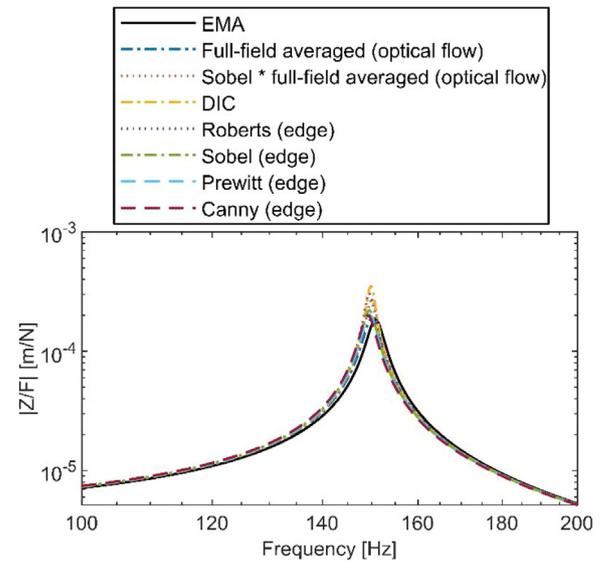


Fig. 9. FRFs of the grooving blade.

amounts of noise reduction – as is indeed observed in Fig. 7(a) for all tools of interest.

From Fig. 7(b) it is evident that the RMS errors for all tools, in general, are lower than the standard implementation of the optical flow-based scheme, results of which are shown in Fig. 7(a). Interestingly however, and unlike the results in Fig. 7(a), the results in Fig. 7(b) show that the RMS errors do not appear to be very strongly correlated to the size of the Gaussian blur. Since increasing sizes of the blur result in greater noise reduction and simultaneously result in a blurry image – which also makes edge detection more difficult, the RMS errors in Fig. 7(b) being relatively independent of the size of the blur are not unsurprising.

Finally, for the kernels and the size of the Gaussian blur that result in lower RMS errors, response comparisons in Fig. 7(c) make it clear that the response obtained by convolving the motion matrices with the Sobel edge detector compare better with the twice integrated measured accelerations than the response obtained by averaging the full-field motion. Furthermore, Fig. 7(c) also shows that the unfiltered response for all tools is clearly corrupted by a 100 Hz peak corresponding to the harmonic of the light flicker frequency. Flicker corrupting measurements is specific to the optical flow-based motion estimation scheme due to its brightness constancy assumption. Since edge detection schemes discussed above and the DIC scheme discussed subsequently, do not make such restrictive assumptions, and nor do they directly use the pixel intensities to register motion, motion estimated using those schemes is not susceptible to being corrupted by flicker. Flicker corrupting measurements can be

Table 1
Modal parameters of the grooving blade.

Method		f_1 [rad/s]	ζ_1 [%]
EMA		$2\pi \times 150.8$	1.08
Edge detection schemes	Roberts	$2\pi \times 149.5$	0.67
	Sobel	$2\pi \times 149.5$	0.92
	Prewitt	$2\pi \times 149.5$	1.00
	Canny	$2\pi \times 149.2$	1.02
	LoG	–	–
Optical flow	Full-field averaged	$2\pi \times 150$	1.07
	Sobel*full-field avg.	$2\pi \times 150$	0.75
DIC scheme		$2\pi \times 149.8$	0.57

Table 2
Modal parameters of the end mill.

Method		f_1 [rad/s]	ζ_1 [%]
EMA		$2\pi \times 612$	2.50
Edge detection schemes	Roberts	$2\pi \times 606.6$	3.10
	Sobel	$2\pi \times 608.2$	2.72
	Prewitt	$2\pi \times 606.5$	2.86
	Canny	$2\pi \times 609.4$	2.59
	LoG	–	–
Optical flow	Full-field averaged	$2\pi \times 601.5$	2.7
	Sobel*full-field avg.	$2\pi \times 604$	2.8
DIC scheme		$2\pi \times 601.6$	2.51

avoided by conducting the experiments under natural lighting conditions, which, in the case of machine tools is not possible. The other remedy would be to use modern fluorescent lamps that instead use an electronic ballast and that operate at significantly higher frequencies (20 kHz – 60 kHz) [41].

The other peak frequencies in the spectra in Fig. 7(c) of the response is consistent with those observed with the edge detection and tracking scheme in Fig. 6(c), and like seen there, herein too the magnitudes of the peaks for different methods of motion extraction are different and this too will likely influence modal parameter extraction discussed in the subsequent main section.

Response obtained from DIC

Motion estimated using the DIC method being influenced by the size of the subset and by the size of the Gaussian blur is characterized by the RMS errors in Fig. 8(a), and for the size of the subsets and Gaussian blurs that result in lower RMS errors, Fig. 8(b) contrasts the motion estimated using DIC with twice integrated accelerations.

From Fig. 8(a) it is evident that the ‘best’ size of the subset that results in a low RMS error (characterized by the colour map) is different for the different tools of interest. Since each tool has different features, and since the pixel resolution for every measurement is different, the subset size that results in lower RMS errors for every tool being different is not surprising. In general, the results show that subset size should not be too large or too small. These notions of small and large are relative to the features of the tool within the regions of interest. Since small sized subsets may not completely encompass the edge of the tool that is to be tracked, and since larger sized subsets will likely include regions outside of the edge of interest that is to be tracked, lower and/or higher sized subsets result in the RMS errors being high. Results in Fig. 8(a) also suggest that the size of the Gaussian blur does not significantly influence the RMS error. This is unlike the results for the edge detection schemes shown in Fig. 6 and/or the optical flow-based schemes in Fig. 7. This analysis hence suggests that the DIC method might be more suitable for applications with noisier images, and that the size of the subset matters more than the size of the Gaussian filter.

From Fig. 8(b), though it appears that the motion estimated using DIC, in general, compares well with twice integrated

measured accelerations, the DIC obtained response for the case of the end mill appears distorted. Furthermore, as already discussed in the context of observations made on results shown in Fig. 5, since the per pixel resolution in the case of the end mill is 20 μm, and since the decayed motion is possibly less than 20 μm, and since we do not use any subpixel level interpolation schemes, there are obvious errors in estimating the low amplitude response of the end mill. Interestingly, though there are noticeable differences in the response amplitudes, the frequency spectra are comparable to those observed with the edge detection and tracking scheme in Fig. 6(c), and the optical flow-based scheme in Fig. 7(c). Furthermore, although the peak frequencies in the DIC response are like those of the twice integrated response, the peak amplitudes are not, and this may influence modal parameter estimation – as discussed next.

Comparative analysis of modal parameters

Modal parameters extracted from vision-based motion estimates are compared herein with parameters extracted from the traditional experimental modal analysis (EMA) procedures in which the input force and the output acceleration were both measured. All measurements for EMA were processed using CutPRO®’s data acquisition and modal parameter extraction modules [38]. For all vision-based measurements, modal parameters, i.e., the natural frequencies and damping coefficients are estimated using ERA [34]. Our use of ERA involves ordering of the singular values to identify the dominant ‘real’ modes of the system. These singular values are obtained by performing a singular value decomposition of a Hankel matrix constructed from the estimated response. Based on how many modes are selected as being potentially ‘real’, we then reorder the system matrices and construct a truncated observability matrix and a shifted Hankel matrix. On obtaining the discrete system realization of the system matrix, we obtain eigenvalues and eigenvectors of the system matrix. Conversion of the complex modes to real modes, and discrete time to continuous time, results in the natural frequencies and damping ratios. For additional details on our ERA implementation, please see [21,36].

The natural frequencies and damping estimates thus estimated are used to reconstruct tool point FRFs. The eigenvectors necessary to reconstruct FRFs are estimated from the traditional EMA procedure. FRFs are reconstructed from the extracted modal parameters using:

$$h(\omega) = \sum_{q=1}^s \frac{\phi_q^2}{-\omega^2 + 2i\zeta_q\omega\omega_q + \omega_q^2}, \tag{19}$$

wherein ω_q , ζ_q , and ϕ_q are the natural frequencies, damping ratios, and mass-normalized eigenvectors, respectively for ‘s’ modes of interest. Modal parameters and FRFs for all three tools of interest are compared subsequently. For the vision-based measurements, modal parameter comparisons include those extracted from motion estimated using the five different edge detection and tracking schemes, from the two different optical flow-based methods, and from the DIC method.

Modal parameters and FRFs of the grooving blade

Modal parameters of the grooving blade are listed in Table 1, which also lists parameters extracted from EMA. The eigenvector estimated from EMA that is used to reconstruct the FRFs is $\phi = 1.9$, and the FRFs are shown in Fig. 9. As is evident from Table 1, the natural frequencies estimated from all vision-based methods are in very good agreement with those extracted from EMA. Though natural frequencies compare well, there are significant differences

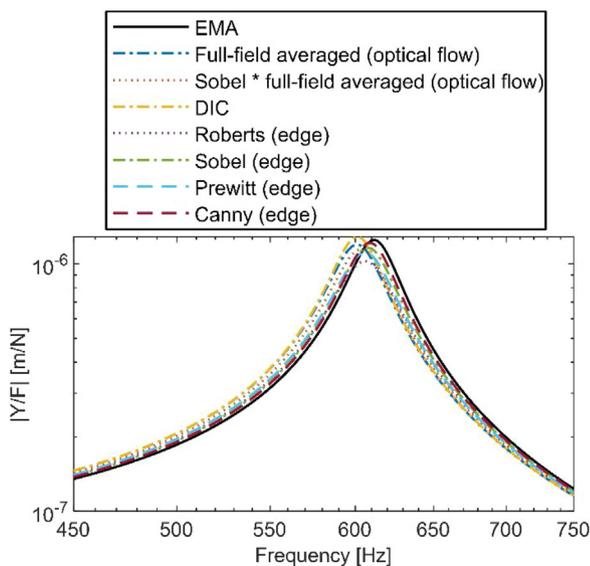


Fig. 10. FRFs of the end mill.

Table 3
Modal parameters of the boring bar for smartphone and high-speed camera measurements.

Method		Smartphone camera		High-speed camera			
		First mode		First mode		Second mode	
		f_1 [rad/s]	ζ_1 [%]	f_1 [rad/s]	ζ_1 [%]	f_2 [rad/s]	ζ_2 [%]
EMA		$2\pi \times 226.5$	0.50	$2\pi \times 226.5$	0.5	$2\pi \times 1235$	0.65
Edge detection schemes	Roberts	$2\pi \times 224.8$	0.80	$2\pi \times 225.3$	0.60	$2\pi \times 1228.4$	0.29
	Sobel	$2\pi \times 225$	0.79	$2\pi \times 225.4$	0.52	$2\pi \times 1231.8$	0.73
	Prewitt	$2\pi \times 224.8$	0.83	$2\pi \times 225.3$	0.71	$2\pi \times 1229.6$	0.73
	Canny	$2\pi \times 225.2$	0.64	$2\pi \times 225.4$	0.56	$2\pi \times 1233.7$	0.45
	LoG	–	–	–	–	–	–
Optical flow	Full-field averaged	$2\pi \times 225.2$	0.26	$2\pi \times 225.4$	0.65	$2\pi \times 1229.2$	0.57
	Sobel*full-field averaged	$2\pi \times 224.9$	0.54	$2\pi \times 225.5$	0.59	$2\pi \times 1228.3$	0.43
DIC scheme		$2\pi \times 225.6$	0.51	$2\pi \times 225.4$	0.42	$2\pi \times 1230$	0.53

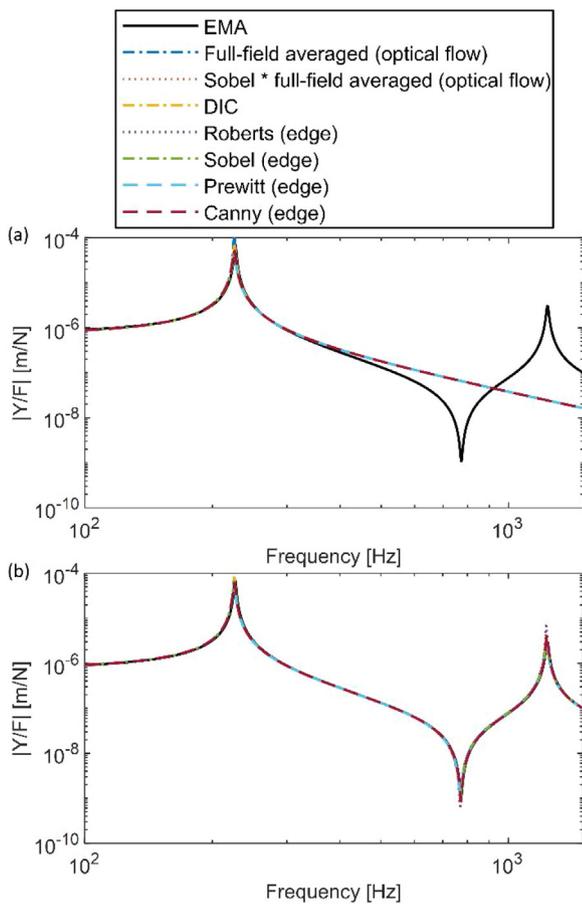


Fig. 11. FRFs for the boring bar with measured with (a) the smartphone camera. (b) the high-speed camera.

in the estimates of damping ratios. The difference in damping in the worst case is $\sim 47\%$ for the use of the DIC method and is $\sim 37\%$ for the case of using Roberts' edge detection scheme. The Canny edge detector and the optical flow-based schemes appear to outperform others. Since response estimated using the LoG edge detector was corrupted (see Fig. 6), no meaningful estimates of modal parameters are possible with the use of the LoG method. The differences in the estimated modal parameters carry forward to the FRFs reconstructed using these parameters shown in Fig. 9. And, as is evident from Fig. 9, the FRFs for the DIC and Roberts' methods are very different than the EMA results. FRFs for the Canny and the optical flow-based scheme agree well with the EMA results.

Modal parameters and FRFs of the end mill

Modal parameters of the end mill are tabulated in Table 2. And even though the response estimated using the different vision-based methods appeared like the twice integrated accelerations (see Fig. 6–8), there are differences in the modal parameters extracted from the different vision-based methods in comparison to those estimated using EMA. And, in the case of the LoG method of vision-based motion registration, since the motion was somewhat distorted (see Fig. 6), no meaningful modal parameters are extractable. Results in Table 2 suggest that the Canny-based edge detection and tracking scheme outperforms the other schemes. However, even in the worst case the maximum difference in natural frequencies between EMA and the vision-based methods is at most $\sim 1.7\%$. The difference in the damping estimate, however, is larger at $\sim 24\%$ in the worst case for the case of the Roberts' method of edge detection. These modal parameters along with the eigenvector estimated from EMA, found to be $\phi = 0.96$, are used to reconstruct FRFs that are shown in Fig. 10. And, though the general trend of all FRFs are comparable, the differences in the modal parameters in Table 2 translate to the differences observed in the FRFs, with the Canny edge detection scheme comparing much better, and the Roberts' scheme faring the worst.

Modal parameters and FRFs of the boring bar

Modal parameters estimated from measuring the boring bar with the smartphone camera as well as with the high-speed camera are both listed in Table 3 along with parameters extracted using traditional EMA measurements. Since the second and higher-frequency mode of the boring bar estimated around ~ 1235 Hz is higher than the acquisition rate of the smartphone camera that records video at 480 fps, this mode is not detectable for recordings with the smartphone camera. Moreover, since the response estimated using the LoG method was corrupted and noisy (see Fig. 6), no meaningful modal parameters were estimated using the LoG scheme.

The natural frequencies of the lower-frequency mode extracted using all vision-based methods compare very well with the EMA results. The damping ratios on the other hand are better estimated from motion estimated from recordings using the high-speed camera than from motion estimated from recordings using the smartphone camera. For the smartphone camera, the difference in the damping ratio of the first mode is at most $\sim 48\%$ for the case of optical flow-based scheme averaging the full-field motion. Similar differences are observed for use of the Roberts' and Prewitt methods of edge detection. For the high-speed camera, the difference in the damping ratio for the Prewitt's method of edge detection is also high ($\sim 40\%$). For the low-frequency mode, the

Canny edge detector along with the DIC method outperforms all other methods.

For the case of the higher-frequency mode, the natural frequency estimates using all vision-based methods are comparable and agreeable with those estimated from EMA. For the case of damping however, there is a ~55% difference between the estimate using Roberts' method and EMA. For this high-frequency mode, the Canny edge detector, both optical flow-based schemes, and the DIC scheme all compare well and agree with results from EMA.

FRFs reconstructed with the modal parameters listed in Table 3 along with the eigenvectors of $\phi_1 = 1.19$ and $\phi_2 = 1.55$ obtained from EMA are compared in Fig. 11. And as is evident, the differences in the estimated modal parameters listed in Table 3 translate to the differences observed in the FRFs shown in Fig. 11. However, since the differences in modal parameters are not very significant (besides the outliers that over/underestimate damping), the FRFs also generally compare and agree well with the results from EMA. The FRFs reconstructed using parameters extracted from recordings using the smartphone camera completely miss the higher-frequency mode – as is evident from Fig. 11(a). For the case of parameters estimated from the high-speed recordings, the FRFs compare better with the EMA results – as shown in Fig. 11(b) than they did for the case of the low speed camera – see Fig. 11(a).

Conclusions

This paper successfully demonstrates the use of vision-based methods for cutting tool motion registration and modal analysis. Motion of three illustrative tools were recorded using low- and high-speed cameras with sufficiently high resolutions. The tool's own features were used to register motion. Three different methods of motion registration were discussed, namely, the edge detection and tracking schemes, the expanded optical flow-based schemes, and established DIC methods, to evaluate their suitability for the application of interest. Extracted motion was benchmarked against twice integrated measured tool point accelerations, and motion was generally observed to compare well. Modal parameters extracted from vision-based measurements were also observed to agree with those extracted using more traditional experimental modal analysis procedures using a contact type accelerometer as the transducer.

Within the class of the five different edge detection schemes investigated herein, the Canny edge detector was observed to consistently outperform other methods and was observed to be less sensitive to thresholds and image noise than the other edge detection schemes. For the expanded optical flow-based scheme presented herein, convolution of the averaged full-field motion with a Sobel operator was observed to perform better than the standard method of implementation. For DIC methods, our analysis reveals that these are more sensitive to size of the subsets correlated than to image noise, and that DIC implementation could benefit from the use of advanced subpixel motion registration algorithms. Since accuracy of motion registration methods presented herein is governed by the object's own features, illumination conditions, noise, and the image acquisition parameters, any of the three methods presented herein could be suitably adapted for any other application of interest.

Our findings may be viewed as a modest first step towards realizing the potential of using vision-based modal analysis as an alternative to the more traditional experimental modal analysis methods using contact type transducers. Furthermore, since we demonstrate that it is possible to extract motion of different tools with different features mounted on different machines using different setups, our methods are not far removed from their intended use in real shop floor environments – which also have these vagaries.

The output-only vision-based methods presented herein can be synchronized to applied forces, in which case, the roundabout method of estimating mass normalized eigenvectors as presented herein can be avoided. Further improvements, as necessary, to capture subpixel level motions are also possible. Moreover, with the rapid developments in camera technologies and with the progress in computer vision techniques, visual vibrometry has the potential to offer alternate paradigms for vibration monitoring and evaluation of machine tool systems.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] Feng, D., Feng, M.Q., 2018, Computer Vision for SHM of Civil Infrastructure: From Dynamic Response Measurement to Damage Detection—A Review. *Engineering Structures*, 156:105–117. <http://dx.doi.org/10.1016/j.engstruct.2017.11.018>.
- [2] Xu, Y., Brownjohn, J.M.W., 2018, Review of Machine-Vision Based Methodologies for Displacement Measurement in Civil Structures. *Journal of Civil Structural Health Monitoring*, 8(1): 91–110. <http://dx.doi.org/10.1007/s13349-017-0261-4>.
- [3] Spencer Jr B.F. 2019, Advances in Computer Vision-Based Civil Infrastructure Inspection and Monitoring. *Engineering*, 5: 199–222. <http://dx.doi.org/10.1016/j.eng.2018.11.030>.
- [4] Chen, J.G., et al, 2015, Modal Identification of Simple Structures with High-Speed Video Using Motion Magnification. *Journal of Sound and Vibration*, 345:58–71. <http://dx.doi.org/10.1016/j.jsv.2015.01.024>.
- [5] Davis, A., et al, 2017, Visual Vibrometry: Estimating Material Properties from Small Motions in Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:732–745. <http://dx.doi.org/10.1109/TPAMI.2016.2622271>.
- [6] Javh, J., et al, 2017, The Subpixel Resolution of Optical-Flow-Based Modal Analysis. *Mechanical Systems and Signal Processing*, 88:89–99. <http://dx.doi.org/10.1016/j.ymssp.2016.11.009>.
- [7] Bebernis, T.J., Ehrhardt, D.A., 2017, High-Speed 3D Digital Image Correlation Vibration Measurement: Recent Advancements and Noted Limitations. *Mechanical Systems and Signal Processing*, 86:35–48. <http://dx.doi.org/10.1016/j.ymssp.2016.04.014>.
- [8] Sarrafi, A., et al, 2018, Vibration-Based Damage Detection in Wind Turbine Blades Using Phase-Based Motion Estimation and Motion Magnification. *Journal of Sound and Vibration*, 421:300–318. <http://dx.doi.org/10.1016/j.jsv.2018.01.050>.
- [9] Baqersad, J., et al, 2017, Photogrammetry and Optical Methods in Structural Dynamics – A Review. *Mechanical Systems and Signal Processing*, 86:17–34. <http://dx.doi.org/10.1016/j.ymssp.2016.02.011>.
- [10] Sutton, M.A., 2009, *Image Correlation for Shape, Motion, and Deformation Measurements: Basic Concepts, Theory and Applications*. SpringerLink. ISBN 978-0-387-78747-3.
- [11] Kurada, S., Bradley, C., 1997, A Review of Machine Vision Sensors for Tool Condition Monitoring. *Computers in Industry*, 34(1): 55–72. [http://dx.doi.org/10.1016/S0166-3615\(96\)00075-9](http://dx.doi.org/10.1016/S0166-3615(96)00075-9).
- [12] Karthik, A., et al, 1997, 3D Tool Wear Measurement and Visualisation Using Stereo Imaging. *International Journal of Machine Tools and Manufacture*, 37:1573–1581. [http://dx.doi.org/10.1016/S0890-6955\(97\)00023-0](http://dx.doi.org/10.1016/S0890-6955(97)00023-0).
- [13] Kumar, R., et al, 2005, Application of Digital Image Magnification for Surface Roughness Evaluation Using Machine Vision. *International Journal of Machine Tools and Manufacture*, 45:228–234. <http://dx.doi.org/10.1016/j.ijmachtools.2004.07.001>.
- [14] Dutta, S., et al, 2013, Application of Digital Image Processing in Tool Condition Monitoring: A Review. *CIRP Journal of Manufacturing Science and Technology*, 6(3): 212–232. <http://dx.doi.org/10.1016/j.cirpj.2013.02.005>.
- [15] Khalifa, O.O., et al, 2006, Image Processing for Chatter Identification in Machining Processes. *The International Journal of Advanced Manufacturing Technology*, 31:443. <http://dx.doi.org/10.1007/s00170-005-0233-4>.
- [16] Szydłowski, M., Powalka, B., 2012, Chatter Detection Algorithm Based on Machine Vision. *The International Journal of Advanced Manufacturing Technology*, 62:517–528. <http://dx.doi.org/10.1007/s00170-011-3816-2>.
- [17] Lei, N., Soshi, M., 2017, Vision-Based System for Chatter Identification and Process Optimization in High-Speed Milling. *The International Journal of Advanced Manufacturing Technology*, 89:2757–2769. <http://dx.doi.org/10.1007/s00170-016-9770-2>.
- [18] Guo, Y., et al, 2015, In Situ Analysis of Flow Dynamics and Deformation Fields in Cutting and Sliding of Metals. *Proceedings of the Royal Society A*, 471. <http://dx.doi.org/10.1098/rspa.2015.0194>. 20150194.
- [19] Berezvai, S., et al, 2018, High-Speed Camera Measurements in the Mechanical Analysis of Machining. *Procedia CIRP*, 77:155–158. <http://dx.doi.org/10.1016/j.procir.2018.08.264>.

- [20] Molnar, T.G., et al, 2019, Experimental Investigation of Dynamic Chip Formation in Orthogonal Cutting. *International Journal of Machine Tools and Manufacture*, 145. <http://dx.doi.org/10.1016/j.ijmactools.2019.103429>.
- [21] Law, M., 2020, Modal Analysis of Machine Tools Using Visual Vibrometry and Output-Only Methods. *Annals of the CIRP*, 69/357: 360. <http://dx.doi.org/10.1016/j.cirp.2020.04.043>.
- [22] Peters, W.H., Ranson, W.F., 1982, Digital Imaging Techniques in Experimental Stress Analysis. *Optical Engineering*, 21/3: 427–431. <http://dx.doi.org/10.1117/12.7972925>.
- [23] Lewis, J.P., 1995, Fast Normalized Cross-Correlation. *Vision Interface. Canadian Image Processing and Pattern Recognition Society*, 120–123.
- [24] Hartmann, C., et al, 2018, Implementation and Evaluation of Optical Flow Methods for Two-Dimensional Deformation Measurement in Comparison to Digital Image Correlation. *Optics and Lasers in Engineering*, 107:127–141. <http://dx.doi.org/10.1016/j.optlaseng.2018.03.021>.
- [25] Horn, B.K.P., Schunck, B.G., 1981, Determining Optical Flow. *Artificial Intelligence*, 319–331. [http://dx.doi.org/10.1016/0004-3702\(81\)90024-2](http://dx.doi.org/10.1016/0004-3702(81)90024-2).
- [26] Lucas, B.D., Kanade, T., 1981, An Iterative Image Registration Technique With an Application to Stereo Vision. *Proceedings of the 7th international joint conference on Artificial intelligence 2*, 674–679.
- [27] Roberts, L.G., 1963, Machine Perception of Three-Dimensional Solids. in *Tippet J.J.T.T. (Ed.) Optical and Electro-optical Information Processing*. MIT Press, pp. 159–197.
- [28] Sobel, I., Feldman, G., 1968, A 3x3 Isotropic Gradient Operator for Image Processing. *A Talk Stanford Artif Proj*, 271–272.
- [29] Prewitt, J.M.S., 1970, Object Enhancement and Extraction. *Picture Processing and Psychopictorics*, 10: 15–19.
- [30] Canny, J., 1986, A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 679–698. <http://dx.doi.org/10.1109/TPAMI.1986.4767851>.
- [31] Marr, D., Hildreth, E., 1980, Theory of Edge Detection. *Proceedings of the Royal Society B: Biological Sciences*, 207:187–217. <http://dx.doi.org/10.1098/rspb.1980.0020>.
- [32] OnePlus, OnePlus 7T, 2019. <https://www.oneplus.in/7t/specs?from=7t>. (Accessed 25 May 2020).
- [33] Photron, FASTCAM SA 1.1. <https://photron.com/fastcam-sa1-1/>. (Accessed 25 May 2020).
- [34] Juang, J., Pappa, R., 1985, An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction. *Journal of Guidance Control and Dynamics*, 8:620–627. <http://dx.doi.org/10.2514/3.20031>.
- [35] Ewins, D.J., 2000, *Modal Testing: Theory, Practice, and Application*. Research Studies Press Ltd..
- [36] Gupta, P., et al, 2020, Evaluating Tool Point Dynamics Using Output-Only Modal Analysis With Mass-Change Methods. *CIRP Journal of Manufacturing Science and Technology*, . <http://dx.doi.org/10.1016/j.cirpj.2020.06.001>.
- [37] Peng, Y., et al, 2018, A Method to Obtain the In-Process FRF of a Machine Tool Based on Operational Modal Analysis and Experiment Modal Analysis. *The International Journal of Advanced Manufacturing Technology*, 95:3599–3607. <http://dx.doi.org/10.1007/s00170-017-1405-8>.
- [38] CUTPRO V11.2. Advanced Machining Simulation Software ©MAL Inc.
- [39] Luo, L., et al, 2018, Robust Vision Sensor for Multi-Point Displacement Monitoring of Bridges in the Field. *Engineering Structures*, 163:255–266. <http://dx.doi.org/10.1016/j.engstruct.2018.02.014>.
- [40] Freudenberg, J.S., 2019, Sampling, Beats, and the Software Oscilloscope. *Lecture notes in Embedded Control Systems*. (Accessed 8 October 2020)In: <https://idsc.ethz.ch/education/lectures/embedded-control-systems.html>.
- [41] Veitch, J.A., McCoil, S.L., 1995, Modulation of Fluorescent Light: Flicker Rate and Light Source Effects on Visual Performance and Visual Comfort. *International Journal of Lighting Research and Technology*, 27/4: 243–256. <http://dx.doi.org/10.1177/14771535950270040301>.