A Study on High-Speed-Vision-Based Vibration Spectrum Imaging

（高速ビジョンに基づく振動スペクトルイメージングに関する研究）

by

Kohei Shimasaki
島崎　航平

Graduate School of Engineering
Hiroshima University
September, 2020

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Vibration is a common problem in various fields such as mechanical system and rotating machinery, several studies using a vibration analysis technology that grasps the target dynamics characteristics from actual measurement data have been conducted [1,2]. In the case of most of the exact measurement, when almost all methods are conducted using sensors such as an accelerometer [3] and a laser Doppler vibrometer (LDV) [4], these methods are the limited number of measurement points in a narrow area or point to point domain because of a spot measurement. These methods can't measure the whole dynamics characteristics of the target object, compared with vibration simulation analysis, such as finite element methods (FEM) [5]. To grasp the whole vibration dynamics by measuring the displacement response, vision-based methods are developed, such as the template macthing [6, 7], optical flow [8], digital image correlation (DIC) [9] and so on. A vision sensor has potential benefits as a non-contact vibration measurement in a wide area, such as numerous points measurement simultaneously. A vision sensor has high spatial resolution and distribution. The vision-based approach of displacement measurements has a limitation in terms of the spatial resolution that the image sensor of a vision determines pixel pitch. Sub pixel-level vibration measurement with laser interferometry using vision-based methods has been conducted such as electronic speckle pattern interferometry (ESPI) [10], as they utilize the interferometric fringe patterns for maginification of small displacements at the laser-wavelength level. Besides, there is a limitation of a low sampling rate, which is not suitable for high-speed invisible phenomena using conventional vision systems mainly. However, they have designed standard video images for human's visible ability which can see something at 30 fps (NTSC) and 25 fps (PAL) although they have a high spatial resolution. On the whole, the high-speed vision has functions of high spatial resolution and high sampling rate which can see invisible phenomena like the acoustic signal at hundreds or thousands of fps. Because a high-frame-rate (HFR) image sensor is regarded as a collection of photosensors that can detect temporal periodic changes in brightness signals. We have proposed a concept of vibration features with vibration features using HFR images for capturing the time-varying changes in an image as a vibration distribution [11–13]. Real-time high-speed vision systems to capture high-speed phenomena that the human eye cannot see have been developed at 1000 fps level [14–17]. With the acceleration of parallel processing on field programmable gate arrays (FPGAs) and graphics processing units (GPUs), various real-time application examples with high frame rates have been reported, such as bridge vibration analysis [18], tracking applications systems [19–21], and robot manipulation [22–25]. High-speed vision has a significant meaning as a dynamic sensing tool which can localize and detect vibration source accurately by utilizing high spatial directivity of the image sensor, if it can extract brightness change of extremely higher frequency level like acoustic signal around vibration source at pixel level than the low frequency of background and so on.

1.2 Purpose of the research

In this study, we develop a vibration spectrum imaging system that implements the pixel-level vibration frequency analysis to capture the time-varying brightness changes in images as a vibration distribution. It realizes dynamic sensing tools for simultaneously

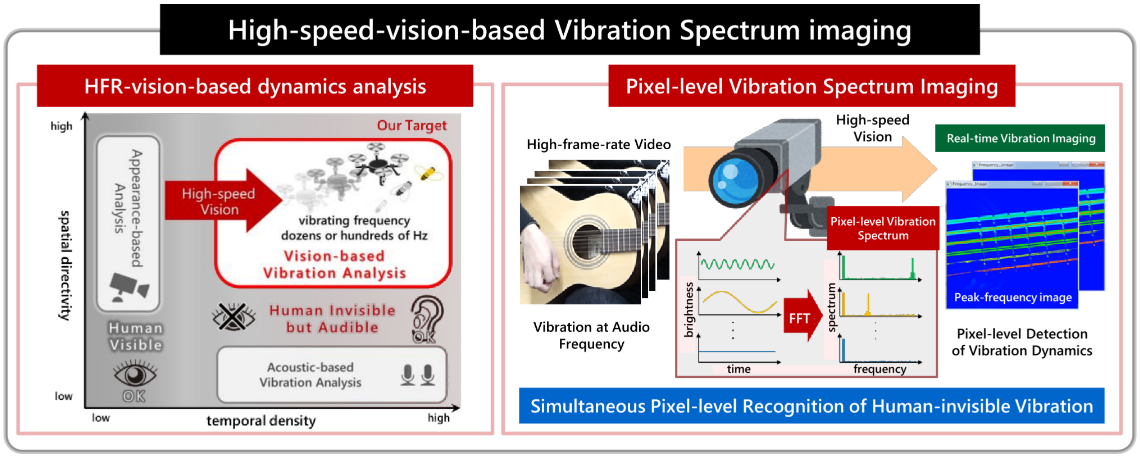

High-speed-vision-based Vibration Spectrum imaging
HFR-vision-based dynamics analysis
high
spatial directivity
low
low
temporal density
high
Our Target
Appearance-based Analysis
High-speed Vision
vibrating frequency
dozens or hundreds of Hz
Vision-based Vibration Analysis
Human Visible
OK
Human Invisible but Audible
OK
Acoustic-based Vibration Analysis
Pixel-level Vibration Spectrum Imaging
High-frame-rate Video
Vibration at Audio Frequency
High-speed Vision
Real-time Vibration Imaging
Pixel-level Vibration Spectrum
brightness
time
FFT
spectrum
frequency
Peak-frequency image
Pixel-level Detection of Vibration Dynamics
Simultaneous Pixel-level Recognition of Human-invisible Vibration

Figure 1.1: Concept overview of this study.

analyzing fast phenomena vibrating at frequencies of dozens or hundreds of Hz, which cannot be seen by the human eye. Our high-speed vision-based vibration analysis target is a human-invisible but audible phenomenon vibrating at an audio frequency level. The vibration spectrum imaging system consists of a high-speed vision used as a sensor for capturing the time-varying changes in an image as a vibration distribution with the acceleration of parallel processing on GPU. We propose novel concepts to realize that a vibration spectrum imaging for the high frequency targets fluctuate at dozens or hundreds of Hz and high-speed moving and vibrating objects.

In our proposed vibration spectrum imaging method, we developed short-time Fourier transform (STFT) functions for the time-varying brightness signals of all the pixels, as we updated a conventional method [11–13] that detects only a specific frequency band with a bandpass filter and didn't apply an unknow frequency detection using a high-frame-rate image. These are used parallel to produce a vibration spectrum imaging that calculates the time-frequency response in real-time for each pixel and realizes to envision the time-varying frequency response of the targets. Figure 1.1 shows the concept of our research to visualize the time-frequency response change in the human audible phenomena.

Our concept is a pixel-level vibration spectrum imaging method that can detect some frequencies in the audio frequency range. This concept was described in Section 3.2

in detail, implemented in the proposed algorithm of Chapter 3. To demonstrate the effectiveness of this concept, several vibration pattern recognition results are reported in Section 3.5, with pixel-level vibration spectrum images for all the pixels of HFR images, in which the fan, multi-copter, and guitar string are rotating or vibrating. The brightness signals for the pixels of their blades and strings fluctuated at dozens or hundreds of Hz. After we verified the effectiveness of visualizing time-varying frequency response of the targets through acoustic signal level examples such as vibrating guitar strings, we made this concept expand two types of actual surveillance and monitoring application for a multi-copter and a honeybee flying in a cluttered outdoor scene, as we utilize a pixel-level vibration source localization method.

In the case of multi-copter, a fast vibration source localization method was described in Chapter 4. Section 4.2 proposes a fast vibration source localization concept using frame-interpolated images and describes our algorithm's details. Section 4.3 outlines the real-time implementation of the algorithm on a high-speed vision system of 512×512 pixels at 1000 fps and evaluates the execution times of the system's pixel-level digital filters. To verify the effectiveness of our concept, Section 4.7 shows the possibility that the performance of the proposed algorithm is almost the same with that of the previous algorithm through we evaluated the experiment for HFR offline video and the real-time experimental results for a flying multi-copter with propellers rotating at 95 rps.

In the case of honeybee, flight activity sensing was described in Chapter 5, and verified its effectiveness through some experiments of flying honeybees in a cluttered outdoor scene. In Section 5.2, we propose a concept of vibration source localization for honeybee activity sensing that executes two-step pixel-level STFTs for an HFR video to obtain the wing-flapping frequencies of fast-flying honeybees as their flight activities, and the details of our algorithm are described. This chapter shows the potential to quantify the wing-flapping frequencies of flying honeybees individually at the audio frequency level.

1.3 Outline of thesis

This thesis is organized as 6 Chapters, including this introduction.

Chapter 2 summarized related works on vibration measurement, high-speed vision and vibration source tracking.

In Chapter 3, a real-time high-speed vision-based vibration spectrum imaging for visualization of time-varying frequency response was developed to verify the effectiveness of our real-time vibration spectrum imaging method. They were demonstrated using the vibration results for a fan, multi-copter, and guitar string, which had rotating or vibrating frequencies in the range of hundreds of Hz.

Chapter 4 explains the fast vibration source localization for a multi-copter using pixel-level digital filters was proposed to verify the effectiveness of pixel-level accuracy in vibration source tracking through several experimental results of offline high-frame-rate videos and real-time implementation, which displayed various flight scenarios of a multi-copter with its propellers rotating at 90-100 Hz.

In Chapter 5, an HFR-video-based honeybee activity sensing was proposed to realize the activity sensing of honeybees flying in a natural outdoor environment by inspecting pixel-level temporal frequency responses in the brightness in an HFR video, computed by executing STFTs of the brightness signals at all pixels. The effectiveness of the proposed algorithm was verified in several experiments which are how the spatial resolution of an HFR video affects the sensitivity in our honeybee activity sensing algorithm and the trajectories and flying activities of honeybees in the outdoor scene.

In Chapter 6, it summarized the contributions of this study and discussed future work.

The appendix included the summary and future implementation of Onsite-Team-Project in the TAOYAKA program which I belonged to became a global leader with a broad perspective and learned onsite reverse innovation and bottom-up approach in a disadvantaged area with plenty of natural resources during master and Ph.D. courses in Hiroshima University.

Chapter 2

Related works

2.1 Vibration measurement

Vibration measurements are effective, reliable, and nonintrusive technologies for monitoring and inspecting various situations such as machine operation, milling process, and structure health monitoring [26–29]. When we need to select appropriate sensors to acquire a precise signal from a target object, we need to care about the possibility of interpreting what the acquired signal indicates. During the vibration transmissions, the vibration produced by a machine component consists of specific frequencies that don't change, although their levels vary from one location to another. Frequency analysis of the vibration signal is widely used to diagnose machine faults [26]. In the case of cutting forces and vibration, we widely utilized these measured parameters to improve their confidence in sensor-based technologies [27]. However, it's difficult for us to detect tool faults because the vibration dynamics of the cutting tool forces were limited. The vibration parameter is detected with a vibration sensor that can acquire an electrical signal converted from these vibration signals by measuring a target body vibration through its mechanical structure. We can analyze this electrical signal for vibration characteristics such as amplitude, frequency, displacement, velocity, acceleration, phase, and period. We can calculate a digital signal processing in the PC environment after converting the analog signal to a digital signal.

Vibration measurements are commonly used such as displacement transducers (vibrometer, proximity probes), velocity transducer (velometer, LDV), accelerometer, vision sensor.

Three types of displacement transducers are generally used for journal bearing equipped rotating machinery for shaft vibration measurement because they preferred to measure the low frequency vibration less than 10 Hz. Firstly, eddy current proximity sensor is used as a non-contact sensor to measure displacement with the electromagnetic induction principle that changes in the resistance of the material. The eddy current proximity sensor that can measure displacement in non-contact methods is used to measure the condition of jet engines [30]. Second, the inductive proximity switch can detect only metal objects such as ferrous metals. When an alternating current passes through the coil, the inductance of the coil changes in present value. It is considered an optimal measurement estimated the rotor-to-stator position [31]. Third, capacitive sensors are non-contact type sensor works on the principle of measuring the electrical capacitance of the gap between the plates. Capacitive sensors are suited for high precision, high resolution, dynamics, and measured vibration amplitude for turbomachinery blades [32].

Velocity transducers are benefitial to use high frequency range of 10 –1000 Hz, compared with displacement transducers. LDV can especially measure the absolute velocity of a vibrating body by utilizing the Doppler shift of scattered laser light from the target object. As LDV utilizes the Doppler shift between the output beam and the reflected beam that is a frequency-modulated laser beam from the target, it can measure the target' velocity. LDV is commonly used for a repeatable and constant vibration measurement such as modal analysis that is a non-destructive process of determining the inherent dynamics properties (natural frequencies, damping ratios, and modal shapes) of vibrating structures [4], because of speckle drop out like laser distortion of speckle noise, although it is of high accuracy and sensitivity.

Accelerometers are electromagnetic transducers to measure vibration acceleration of a target object [3]. The principle is utilizing the piezoelectric effect that the force caused by vibration causes the mass to squeeze the piezoelectric materials which produce an electrical charge that is proportional to the force exerted upon it. Accelerometers can measure the vertical vibration of a target object in a various field such as tool condition monitoring [33,34]. Although they can measure high frequency range with a compact and light-weight body, they need to use amplifier for magnification of a weak input signal.

Vision sensor has potential benefits as a non-contact vibration measurement can capture a high spatial resolution image at the same time, such as numerous numbers of points measurement, as above sensors are the limited number of measurement points in a narrow area because of a spot measurement. A vision sensors can measure displacement and velocity by utilizing digital signal processing such as the template matching [6, 7], optical flow [8], digital image correlation (DIC) [9] to grasp whole vibration dynamics. Vision sensors have a limitation of the measurable frequency range in terms of low sampling rate because input framerate is widely used at 30 fps (NTSC) and 25 fps (PAL).

2.2 High-speed vision

Conventional vision systems mainly elements standard video images for human's visible ability which can see something at 30fps (NTSC) and 25fps (PAL) and have been applied to various fields, such as multimedia, industry, traffic system, biomedical, and so on. Many appearance-based features, such as scale-invariant feature transform [35], Haar-like features [36], histogram of oriented gradients [37], and local binary patterns [38], which can be easily recognized by the human eye, have been used for target tracking in various applications, such as traffic monitoring [39], human-computer interaction [40], and so on. These systems are not suitable for high-speed invisible phenomena such as cell-inspection, factory automation, and robot manipulation and so on, because of the low framerate. Real-time high-speed vision systems to capture high-speed phenomena that the human eye cannot see have been developed at 1000 fps level [14–17]. We have already shown the validity for high-speed tracking using high-speed vision systems such as optical flow [41], camshift tracking [42], feature point tracking [43] and face tracking utilizing Haar-like feature [44] with the acceleration of parallel processing on field programmable gate arrays (FPGAs) and graphics processing units (GPUs). Laser triangulation [45, 46] and LDV [47] as vibration detection using optics sensors have been proposed. Still it is not suitable for these to measure the spatial frequency distribution simultaneously because these approaches are mainly designated for micro-displacement measuring of one point, such as spot inspection. On the while, in the case of high-speed vision, it can drastically improve the amount of processing information per time. It can be visual sensing which can give us new quality from the point of view that it can recognize and quantify high-speed phenomena human can't see. Mainly, it has the significant validity of a dynamic sensing tool that can see the repetitive operation and phenomenon vibrating at sound frequency level at dozens or hundreds of Hz as time-space distribution. And it has shown the effectiveness of cell sensing in a microchannel [48, 49] and vision-based structural modal analysis [50]. High-speed vision has a great meaning as a dynamic sensing tool which can localize and track vibration source accurately by utilizing high space directivity of the image sensor, if it can extract brightness change of extremely higher frequency level like acoustic signal around vibration source at pixel level than the low frequency of background and so on.

2.3 Vibration source localization

Numerous source localization methods have been proposed [51] to simulate the human auditory sense. The binaural approach utilizes left-right-differentiated acoustic features such as interaural level difference (ILD) and interaural time difference (ITD) [52–54] corresponding the human's left and right ear. The microphone-array approach such as stereo auditory sense utilizing interaural time-intensity difference [55], correlative peak calculation in differential arrival time distribution based on multi-microphone array system [56–58], MUltiple-Signal-Classification (MUSIC) methods [59, 60] utilizing quadrature component in a subspace of signal and noise, and beamforming methods [61, 62] emphasizing the directivity for designated direction, can improve localization performance and robustness by using multiple microphones organized along with a spatial geometry. They have been used in many applications such as multi-speaker recognition in human-computer interaction [63], mobile robot audition [64], and operational vibration surveillance in product machines [65], automobiles [66], and aircraft [67]. And the open-access software of HARK [68] that consisting of sound source localization modules, sound source separation modules, and automatic speech recognition modules has been used widely, as they realize their actual application in computational auditory scene analysis. Also, the many applications related to monitoring vibration [69–71] by introducing sound source localization for the manufacturing line, automobile, and plane at work have also been reported because we want to specify vibrated abnormal parts of those machines. Following the recent popularization of multi-copter technologies, audio-signal-based anti-multi-copter surveillance systems [72–75] can extract the sound drone-propellers emit while rotating at dozens or hundreds of Hertz, have been developed to detect illegal or abnormal drones. However, because of the low directivity of sound propagation, the localization accuracy of these methods is limited still when microphones are distant from source objects.

Considering that a multi-copter's propellers rotating at dozens or hundreds of Hertz are observable in an HFR video recording at 1000 fps or more, vision-based vibration source localization methods [12, 13] have been proposed for multi-copter and honeybee tracking by executing digital bandpass filters for brightness signals at all pixels; they can localize the pixels around objects vibrating at frequencies in the audio frequency range without considering their appearances.

These vision-based methods can realize vibration source localization by detecting the time-space change of image intensity as vibration distribution around vibration source. This is because the image sensor is an aggregation of photodetector and image intensity of each pixel. It is time sequential signal regarding brightness change in time space and all the pixels in the image can pass the image intensity signal in a specific frequency band by implementing pixel-level digital filter. This strong point is that it can realize the pixel-level vibration source localization only by using digital filter without implementing the recognition algorithm based on the spatial pattern. It is a valid method in the special case that the frequency of multi copter's propeller is much higher than one of the backgrounds. A flying multi-copter was used to demonstrate the robustness of the methods under the appearance change of occlusion, scale, focus blur, illumination condition, complex background, occlusion, scale, position, posture, and decrease resolution's quality using low spatial resolution's camera. Their temporal frequency responses (TFRs) in brightness signals have been used for pixel-level vibration pattern recognition with a specific vibration pattern [76]. Since the wing-flapping frequencies of flying honeybees are similar to the rotation frequencies of propellers of flying multi-copters, pixel-level digital signal processing for HFR videos could be utilized for localizing flying honeybees. Their wing-flapping frequencies could be obtained as their flight activities even when their appearances were sparse in the low resolution and blurred images.

Chapter 3

Real-time high-speed vision-based vibration spectrum imaging

3.1 Introduction

Several studies that utilized vibration information for inspections in various fields have been conducted in recent years [77–79]. Two main methods have been used to detect vibration displacement: contact measurement and non-contact measurement. In contact measurement, contact sensors such as accelerometers [80] are directly installed on the target objects' vibrated conditions. It is necessary to attach the sensor to the reference point on the structure when attaching a contact sensor. However, it is often difficult or impossible to access the structural points at which it is installed. In non-contact measurement, the vibration displacement of the object to be measured is directly measured with sensors such as an eddy current sensor [81], an optical fiber sensor [82], a laser Doppler vibrometer [83], and an offline-based high-speed vision system [84]. A beneficial vibration analysis of the time-varying data measured by optical sensors in the frequency domain can then be conducted after remotely measuring the vibrated conditions of the target objects. Especially in the case of engineering, it is necessary to measure the vibration displacement [85]. In the case of a machine tool system, low vibration and noise equipment are required for all kinds of machinery. Besides, it is always necessary to monitor and analyze the running condition of a machine during its operation [86]. It is essential to perform vibration analysis and vibration design of mechanical structures to improve their vibration resistance. Therefore, vibration detection with non-contact sensors is more important than contact sensors when we measure and monitor vibration information simply.

Although we can see high-speed vision system as a dynamic sensing tool in subsection 2.2, various types of dynamic sensing applications have been reported using high-speed vision systems, such as the scratching behavior analysis of laboratory mice [87], cell deformation analysis using microchannels [88, 89], and vision-based structure analysis [90, 91].

Because a high-frame-rate (HFR) image sensor is regarded as a collection of photosensors that can detect temporal periodic changes in brightness signals, we have proposed a tracking algorithm with vibration features using HFR images [11]. A pixel-level vibration source localization method that uses pixel-level digital filters [12, 13] has been proposed. A high-speed vision system is used as a sensor for capturing the time-varying changes in an image as a vibration distribution. This method used multi-copter tracking as an example in subsection 4.1 and utilized the fact that the frequency band of the multi-copter propeller was extremely high compared to the low-frequency band, which included the background and cyclic time series brightness signals of the propeller region. Moreover, the presence or absence of a signal was detected using a bandpass filter with a center frequency corresponding to the rotation speed of the propeller for all the pixels. Also, when we design to determine the target frequency band in this method, we need to know the target frequency beforehand. Therefore, it was difficult to analyze a vibrating object when there were large frequency fluctuations or a scene with an unknown vibration frequency such as abnormal vibration.

In this vibration spectrum imaging method, we developed short-time Fourier transform (STFT) functions for the time-varying brightness signals of all the pixels, as we renewed a conventional method [12, 13] that detects only a specific frequency band with

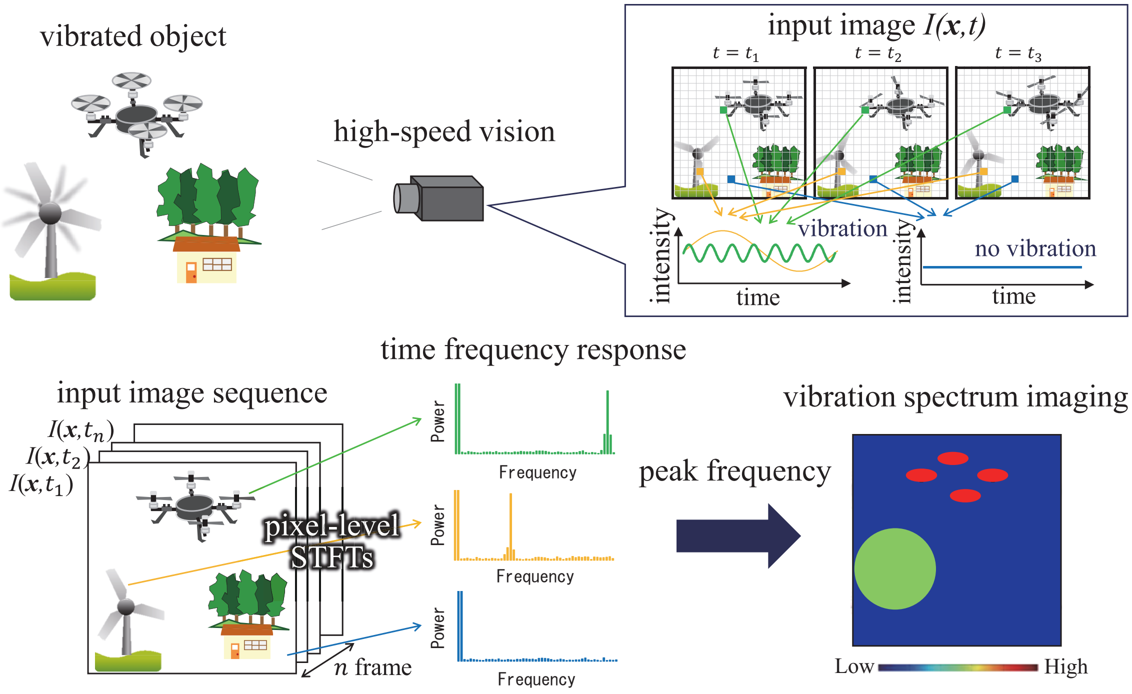


Figure 3.1: Concept of pixel-level vibration spectrum imaging method

a bandpass filter, based on a high-frame-rate image acquired with a high-speed vision system. These are used in parallel to produce a vibration spectrum imaging system that calculates the time-frequency response in real-time for each pixel and realizes to visualize the time-varying frequency response of the targets.

3.2 Concept

The dynamic property of a vibrating object is expressed as its frequency response, which can be obtained by converting the time-varying signal measured by an audio or vibration sensor into a frequency in the frequency domain using the Fourier transform in the majority of sound and vibration analyses. When a vision system operates at a sufficiently high frame rate, temporal periodic changes can be observed in the brightness signals at the pixels around vibrating objects corresponding to their vibration dynamic properties in the audio frequency range. The brightness signals' dynamic features can be quantified in the form of a pixel-level distribution by converting the brightness signals for all the pixels of the HFR images into their temporal frequency responses using the Fourier transform. On the premise of a high frame rate image, the proposed concept related to the vibration spectrum imaging based on the STFT implementation for the time-varying signal of all the pixels is shown in Figure 4.1. When the STFT sampling rate corresponds to the high-speed vision system's frame rate, the STFT can be used for vibration analysis in a frequency range of less than half the frame rate. For example, when high-speed vision at 1000 fps is used, it is possible to conduct a vibration analysis in the frequency band of 0-500 Hz. The processing flow in pixel-level vibration spectrum imaging with the final objective of peak frequency detection is as follows.

3.3 Proposed algorithm

(a) Acquisition of input HFR images

The input HFR images are acquired at time $k\Delta t$ (frame number k) as $I(x, y, k\Delta t)$, wherein the frame rate is $f_0 = 1/\Delta t$.

(b) Computation of STFT images

For the input HFR images of K frames, the brightness signal at pixel (x, y) at time $k\Delta t$ is converted to its TFR $\boldsymbol{F}(x, y, t)$ using the STFT as follows:

$$\begin{aligned} \boldsymbol{F}(x,y,t) &= (F_0(x,y,t),\cdots,F_{K\text{-}1}(x,y,t)), \\ &= \mathrm{STFT}(I(x,y,k\Delta t),\cdots,I(x,y,(k+K-1)\Delta t). \end{aligned} \tag{3.1}$$

(c) Cut-off of low-frequency components

Low-frequency components of less than $f_{k_0} = f_0 k_0/(2K)$ are cut-off from the STFT image $\boldsymbol{F}(x, y, t)$ as follows:

$$\boldsymbol{F}'(x, y, t) = (0, \cdots, 0, F_{k_0}(x, y, t), \cdots, F_{K-1}(x, y, t)). \tag{3.2}$$

(d) Normalization of cut-off STFT image

The cut-off STFT image $\boldsymbol{F}'(x, y, t)$ is normalized as follows:

$$\tilde{\boldsymbol{F}}'(x, y, t) = \boldsymbol{F}'(x, y, t)/|\boldsymbol{F}'(x, y, t)| \tag{3.3}$$

(e) Computation of peak-frequency

We detect the frequency at which the spectral component is the maximum in the frequency band, excluding the DC component in (c) as the peak frequency for the power spectrum calculated for each pixel.

$$\boldsymbol{P}(x, y, t) = \frac{f_0}{2K} \cdot \text{arg max}\ \tilde{\boldsymbol{F}}'_k(x, y, t) \tag{3.4}$$

3.4 Implementation for GPU-based high-speed vision

It is difficult to realize a real-time software implementation of the above algorithm for high-speed vision using only a CPU because the number of STFT calculations for all the pixels becomes increasingly significant. Therefore, in this study, we decided to implement the real-time vibration spectrum imaging function using a GPU-based high-speed vision system that combined general-purpose computing on a graphics processing unit (GPGPU) board and a high-speed camera that enabled parallel implementation of the proposed algorithm. This GPU-based high-speed vision system consisted of a high-speed vision IDP Express [15], GPGPU board (Geforce GTX 1080Ti Blower, ZOTAC), and Hewlett Packard PC (Z440 Workstation, (3.60 GHz, 6 cores, 15 MB, 2.4 GHz), Intel C612 chipset, 32 GB memory, Windows 7 Professional OS (64 bit)). Besides, the IDP Express consisted of the camera head and IDP Express board. This camera allowed 8 bit gray-scale 512×512 images to be captured and processed at a maximum of 2,000 fps, and the processed results were transferred to the PC memory at high speed. The Geforce GTX 1080 Ti (NVIDIA) on the ZOTAC Geforce GTX 1080Ti Blower had 3584 core proces-

Table 3.1: Execution times on a PC.

Image size	128×128	256×256	512×512	1024×1024
K = 64	43.2 ms	373.2 ms	1854.6 ms	7434.8 ms
K = 128	117.7 ms	932.7 ms	4381.5 ms	17145.6 ms
K = 256	350.1 ms	2280.3 ms	9988.3 ms	40039.8 ms
K = 512	1028.9 ms	5178.1 ms	21860.8 ms	84388.7 ms

sors operating at a basic operating frequency of 1480 MHz, as well as 11 GB of GDDR 5X global memory with a bandwidth of 484.4 GB/ s, which provided the computational performance needed to execute floating point arithmetic with 11.3 Tflops. We evaluate the execution times of our algorithm using the personal computer (PC) and GPU to calculate the subprocesses (a)–(e) in the pixel-level vibration spectrum imaging. Table 3.2 summarizes our algorithm's execution times for various image sizes, where K indicates the number of frames in the STFT computation. Our algorithm's execution time increased in proportion to the total number of image pixels and the number of images in the STFT computation. In the case of the real-time software execution, the operable frame rates of the vision system were 840, 206, 56, and 13 fps for images with sizes of 128×128, 256×256, 512×512, and 1024×1024, respectively, when K = 128. The software could process the low-resolution images in real-time at several frames per second. In contrast, our algorithm should be accelerated for the real-time processing of the higher resolution images at dozens of frames per second by implementing the parallel processing logic of our algorithm on specific accelerators such as general-purpose GPUs. When we implemented a pixel-level STFT calculation for K = 128 frames using 512×512 images at 2000 fps, the execution could be performed in 17.89 ms. It was possible to conduct all of the processes from the pixel-level STFT calculation to the peak frequency image output. We confirmed that the vibration spectral image could be displayed in real-time, including a series of operations, at 50 fps or higher.

Table 3.2: Execution times on PC with GPU.

Image size	128×128	256×256	512×512	1024×1024
$K = 64$	0.66 ms	2.20 ms	7.60 ms	31.06 ms
$K = 128$	1.19 ms	4.85 ms	17.89 ms	74.08 ms
$K = 256$	2.55 ms	10.92 ms	42.06 ms	173.90 ms
$K = 512$	5.69 ms	24.50 ms	95.93 ms	391.85 ms

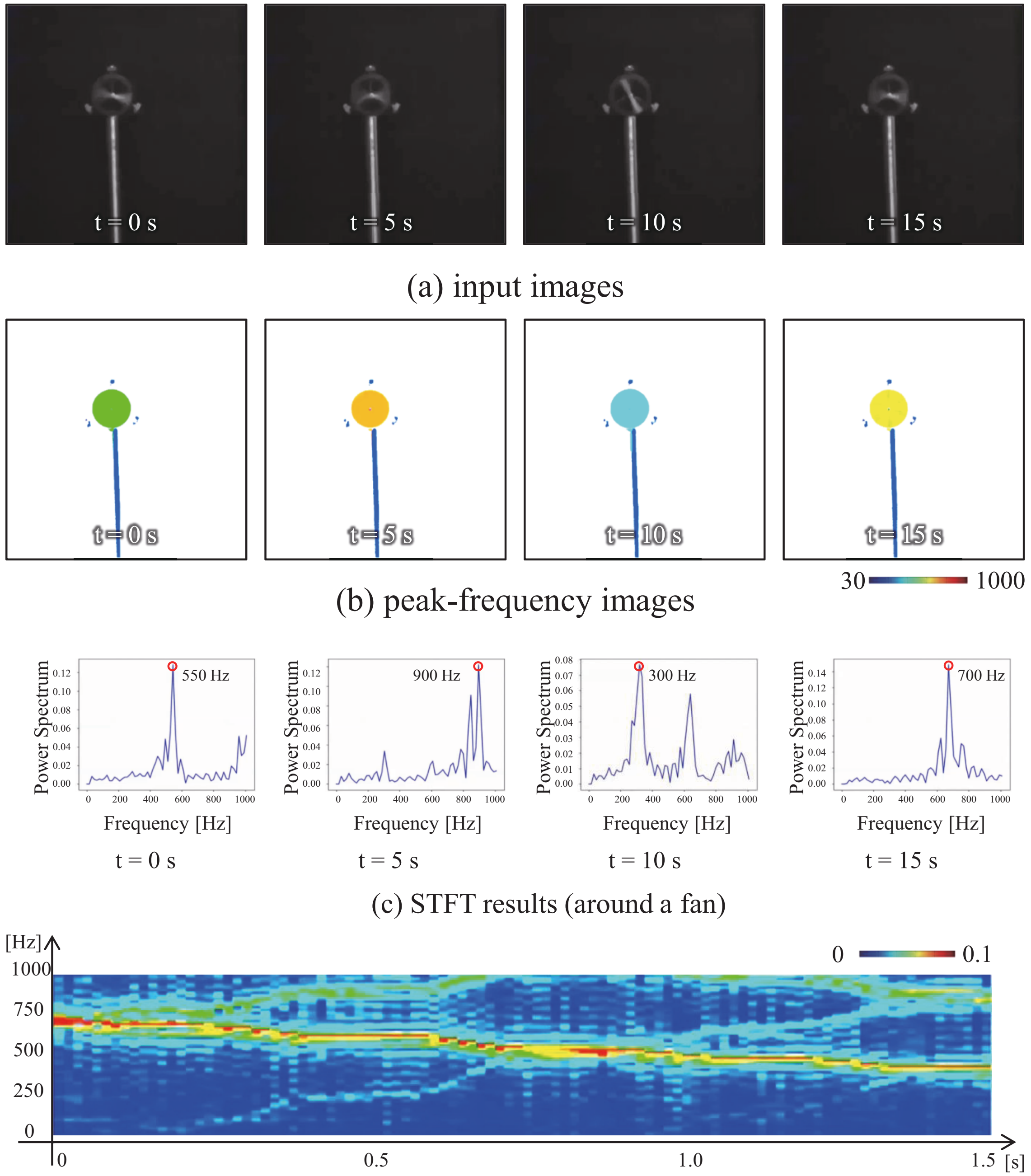
3.5 Real-time vibration spectrum experiments

3.5.1 Fixed fan with variable rotation speed

First, we conducted a real-time vibration imaging experiment using 512×512 images at 2000 fps for a fixed fan whose blade rotation speed fluctuated. In the center of the image, we fixed a fan with four 5 mm long two-bladed propellers. We then changed their speed to 275, 450, 150, and 350 rotations/second at time $t = 0$, 5, 10, and 15 s, respectively. We considered 128 frames ($K = 128$) in this experiment when computing the STFT operations;the cut-off frequency was $f_{k_0} = 30$ Hz. Figure 3.2 shows the (a)input image; (b) peak-frequency image, (c) STFT results for $t = 0$, 5, 10, and 15 s; and (d) vibration frequency spectrum for a 1.5 s period at 15 s. We can see peak frequency fluctuated from vibration frequency spectrum around a fan in (c) and (d). Based on these results, even when the target vibration frequency fluctuated, a peak frequency corresponding to twice the rotational speed of a fan was detected, and pixel-level vibration spectrum imaging was realized in real-time.

3.5.2 Operation estimation for fixed multi-copter based on the rotational speed of each propeller

We conducted a real-time vibration imaging experiment using 512×512 images at 500 fps for a multi-copter whose propeller rotation speed fluctuated. In the center of the image, we fixed a multi-copter (RC EYE One Xtreme, CEI Conrad), 225×225×80 mm



(a) input images

(b) peak-frequency images

(c) STFT results (around a fan)

(d) vibration frequency spectrum for a 1.5 s period at 15 s

Figure 3.2: Vibration spectrum imaging for a fixed fan

in size, with four 138 mm long two-bladed propellers. At time t = 0, 5, 10, and 15 s, the propeller operation was changed to hovering, moving forward, turning left, and turning right. In the experiment, the STFT computation was conducted for K = 128 frames;

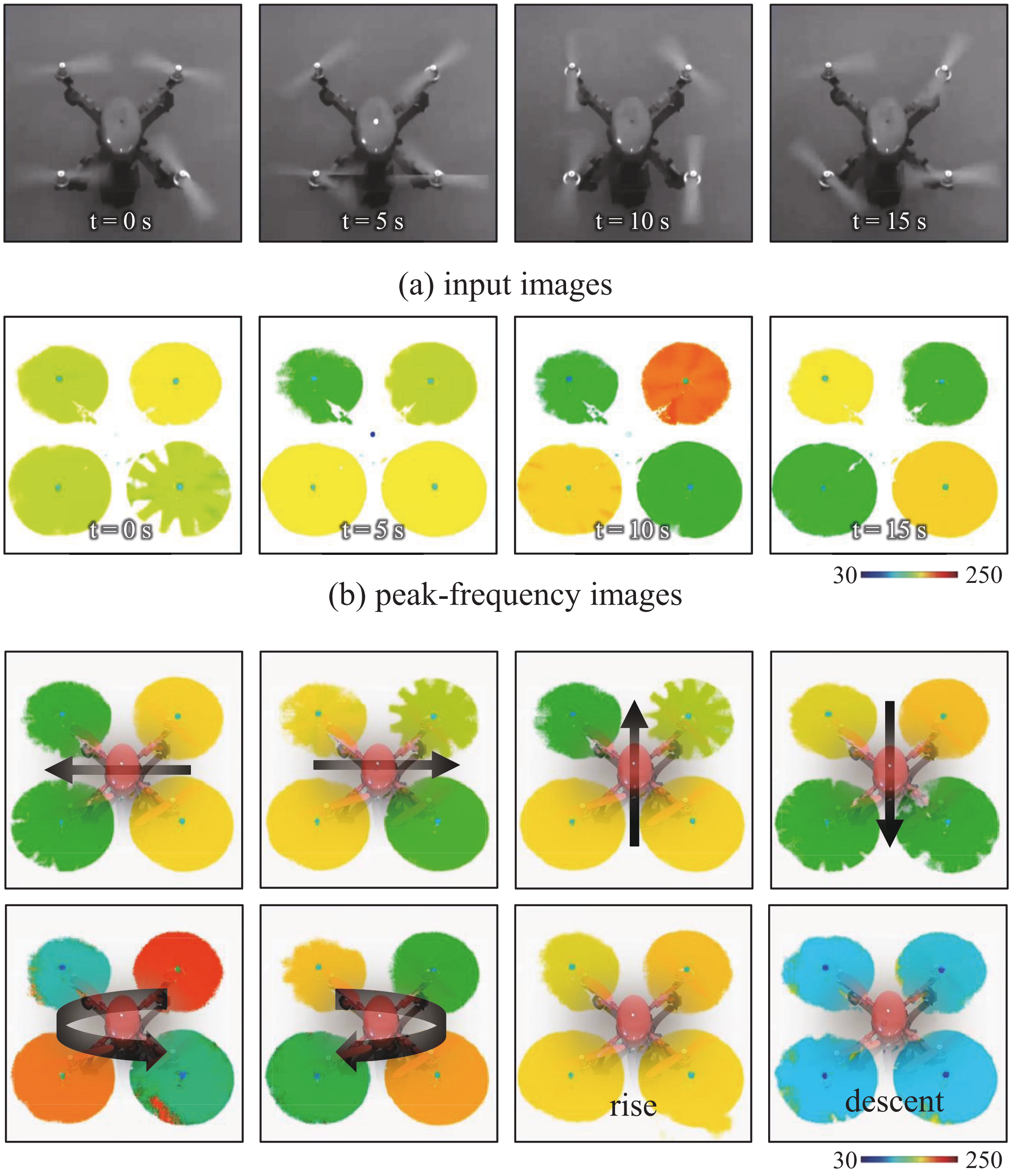
the cut-off frequency was $f_{k_0} = 30$ Hz. Figure 3.3 shows the (a)input image; (b) peak-frequency image for $t = 0$, 5, 10, and 15 s; and (c) moving direction classification based on the peak frequency. These results could be used to estimate the moving direction of the multi-copter from the top view, and pixel-level vibration spectrum imaging was realized in real-time.

3.5.3 Guitar string vibration

We conducted a real-time vibration imaging experiment with a vibrating open guitar string using 512×512 images at 1000 fps. We considered 128 frames ($K = 128$) in this experiment when computing the STFT operations; the cut-off frequency was $f_{k_0} = 0$ Hz. Figure 3.4 shows the (a)input image and (b) peak-frequency image for $t = 0$, 5, 10, and 15 s. These results indicate that peak frequencies of approximately 82, 110, 147, 196, 247, and 330 Hz, corresponding to the frequencies of the 6th to 1st strings, were sequentially detected from the top, and real-time imaging was realized for objects with vibrations at the audio frequency level.

3.6 Concluding remarks

In this chapter, we developed a real-time vibration spectrum imaging system in which STFT functions were implemented parallel for the time-varying signals of all the pixels. This camera allowed vibration imaging at 50 fps or higher using 512×512 images at hundreds and thousands of frames per second to be realized in real-time. Also, we demonstrated its effectiveness through verification experiments using several vibrating objects. We are planning to research mechanical vibration analysis, structural inspection, and biological dynamics inspection as applied research on real-time vibration spectrum imaging.

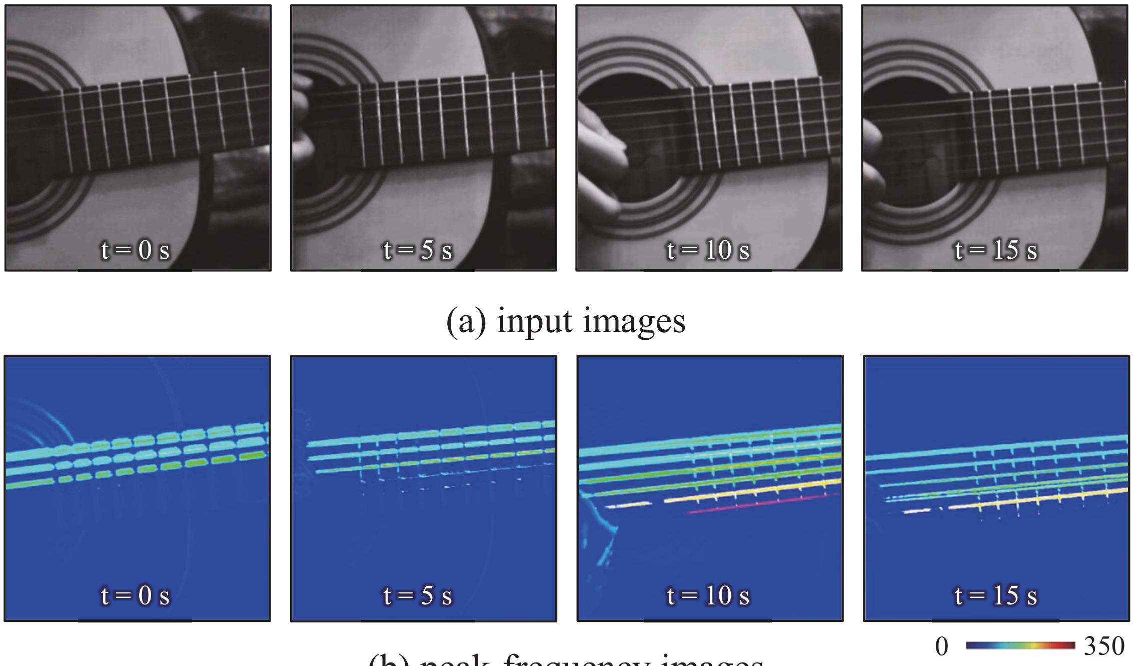


(a) input images

(b) peak-frequency images

(c) moving direction classification based on the peak frequency

Figure 3.3: Vibration spectrum imaging for a fixed multi-copter



(a) input images

(b) peak-frequency images

Figure 3.4: Vibration spectrum imaging of guitar strings

Chapter 4

A fast vibration source tracking algorithm using pixel-level digital filters for a flying multi-copter

4.1 Introduction

With the recent decrease in the price of multi-copters, the use of drone technology has expanded globally. Drone technology has been applied in various aerial photography applications for the press, entertainment, infrastructure maintenance, and disaster monitoring. It is predicted that a vast worldwide market for drone technology will emerge in the next decade, including many aerial services such as delivery drones to transport packages and wireless communication with drone-based mobile stations as well as aerial photography. However, with the expansion of drone technology, the number of untoward or unfortunate incidents with multi-copters caused by out-of-control crashes, terrorism, smuggling, and privacy invasion has rapidly increased. These illegal and immoral intrusions with multi-copters have become a severe social problem when promoting drone technology [92].

As a countermeasure against such illegal intrusions by multi-copters, there is a fast-growing demand for surveillance technologies that can detect, localize, and identify multiple types of multi-copters flying in the air and regulations for air traffic control and operations involving multi-copters. Acoustic signal-based drone monitoring systems that can identify the sound at dozens or hundreds of Hz generated by the rotating propeller of a flying multi-copter have been developed [72–75]. There have been several systems introduced for drone monitoring at airports and infrastructures in the European Union and the United States of America. These acoustic-based systems can detect the presence of a multi-copter by acoustic signal processing. However, they remain inaccurate in localizing and tracking a multi-copter when they are flying outdoors at a long distance from the acoustic sensors because of the very limited spatial directivity of acoustic sensors.

Optical image sensors have an advantage over acoustic sensors with their acute spatial directivity compared with acoustic sensors. Moreover, many vision-based target tracking systems that can accurately localize a target object in images have been developed for robot vision [93]. These systems used conventional image sensors with standard video signals at dozens of frames per second and were suitably designed for the naked human eye. Many appearance-based features, such as scale-invariant feature transform [35], Haar-like features [36], histogram of oriented gradients [37], and local binary patterns [38], which can be easily recognized by the human eye, have been used for target tracking in various applications, such as traffic monitoring [39], human-computer interaction [40], and so on. To detect flying objects, Rozantsev et al. [94] proposed an integrated method that utilized both motion estimation from image sequences and appearance-based features. It is difficult to detect time-varying brightness signals at hundreds of Hertz in image sequences with dozens of frames per second, which correspond to the rotating propellers of a multi-copter. The appearance-based method has limitations in tracking accuracy and robustness when the appearance of a flying multi-copter changes according to the lighting condition, background, occlusion, scale, and so on.

As we introduced a pixel-level vibration source localization method without any knowledge of its appearance features in subsection 3.1, Jiang et al. [12, 13] proposed this method to extract time-varying brightness changes with digital filters at all the pixels in HFR images. However, its computational complexity was too high for the real-time execution of digital filters at all the pixels in the high image resolution of HFR images. They succeeded in implementing ROI image tracking in real-time, as they utilized small resolution in a limited area, compared with input resolution because of the limitation of execution time. Thus, we propose a novel fast vibration source tracking algorithm that can reduce the computational complexity of pixel-level digital filters for HFR videos by virtually adjusting the sampling rates of the digital filters to twice the target's vibration frequency with frame interpolation through sampling rate locks in the object's frequency. Then we also developed a real-time multi-copter tracking system on which pixel-level digital filters can detect vibration distributions around the rotating propellers of a flying multi-copter. Finally, we realized that high-speed and high spatial resolution camera could speed up. We need to implement the digital fiter regarding the higher resolution to monitor vast areas.

4.2 Concept

Assuming that the frame rate of a vision system is a much higher than the target's vibration frequency, and its image sensor is a collection of photodetectors, Jiang's method [12] brought the realization that vibration source localization by extracting the specific frequency components of brightness signals at pixels around vibrating objects with pixel-level digital filters. Figure 4.1 illustrates this concept. The advantage of this concept is that the simple implementation of digital filters on all the pixels enables vibration source localization without appearance-based recognition. This method is effective in localizing a vibrating object, whose vibration frequency is much larger than that of its background scene, such as a flying multi-copter with rotating propellers, because of its robustness to time-varying appearances that depend on lighting conditions, lens blur, scale, and posture changes, and occlusion.

Real-time multi-copter tracking with pixel-level digital filters has been reported for

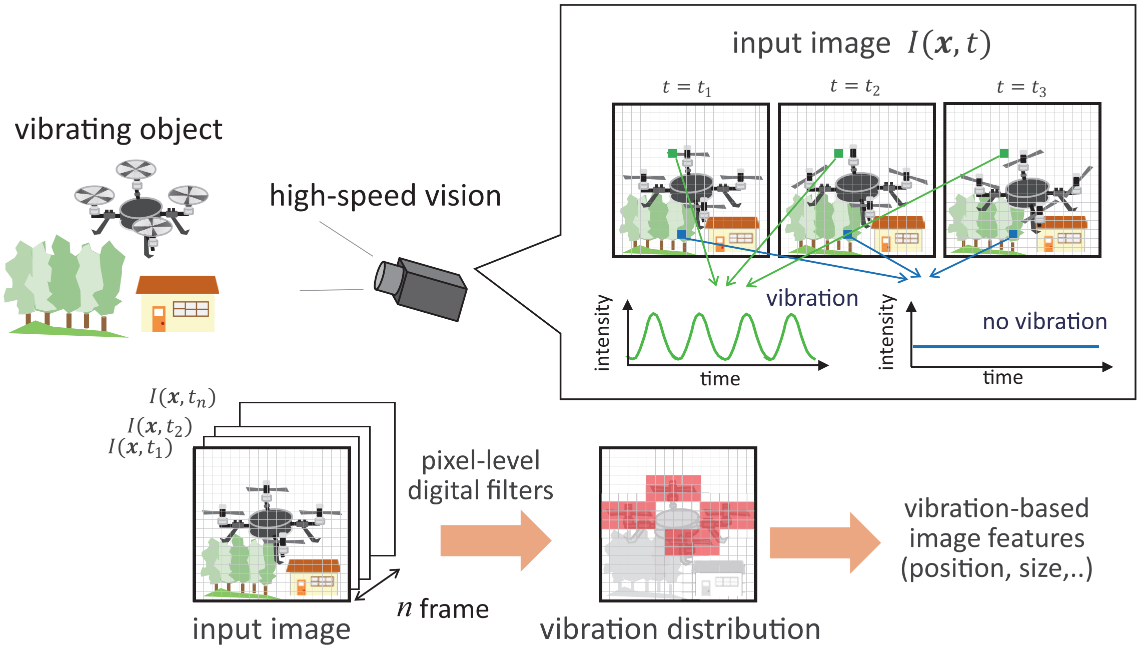


Figure 4.1: Concept of vibration source localization method with pixel-level digital filters

images at 1000 fps [13]. In contrast, the number of pixels on which digital filters were implemented for real-time execution has been limited due to its computational complexity. Digital filters were implemented at 128×128 pixels selected from 512×512 input images in [13]. Thus, it is crucial to accelerate the speed of a vibration source localization algorithm that simultaneously executes digital filters at much more pixels in HFR images. This method enables real-time tracking with much higher spatial resolution of a multi-copter flying in wide-view scenarios.

4.3 Proposed algorithm of vibration source tracking algorithm using frame interpolation

In general, we can design the digital filter passing a specified vibration frequency component if the object frequency is lower than the Nyquist frequency, half of the framerate. On the while, the computational complexity of the proposed method [11–13] depends on the framerate. It calculates the bandpass filter, which refers to center frequency and object frequency by each pixel for video image recording at fixed framerate. This is why it is not necessarily efficient for the previous method to calculate extra complexity in the case of tracking objects whose frequency is much lower than the framerate.

We propose the high-speed vibration source tracking algorithm which reduced the complexity per unit time by implementing the pixel-level high pass filter that can pass the frequency component around target frequency. After we set framerate twice as many as target frequency using downsampling by frame interpolation, assuming we have already known the target frequency, which is tracking the object's frequency. Figure 4.2 shows the flowchart of this method comparing the previous method. We selected the Linear frame interpolation method, which can equalize the frame period, virtually not selecting the Nearest neighbor interpolation method, which has the possibility of a frame period being irregular. It has the purpose of accurately detecting the frequency component by using a digital filter, assuming the sampling period is uniform. The detailed processing flow of this proposed algorithm will show from now. The vibration feature to be evaluated in this study is calculated as follows:

(1) Image acquisition

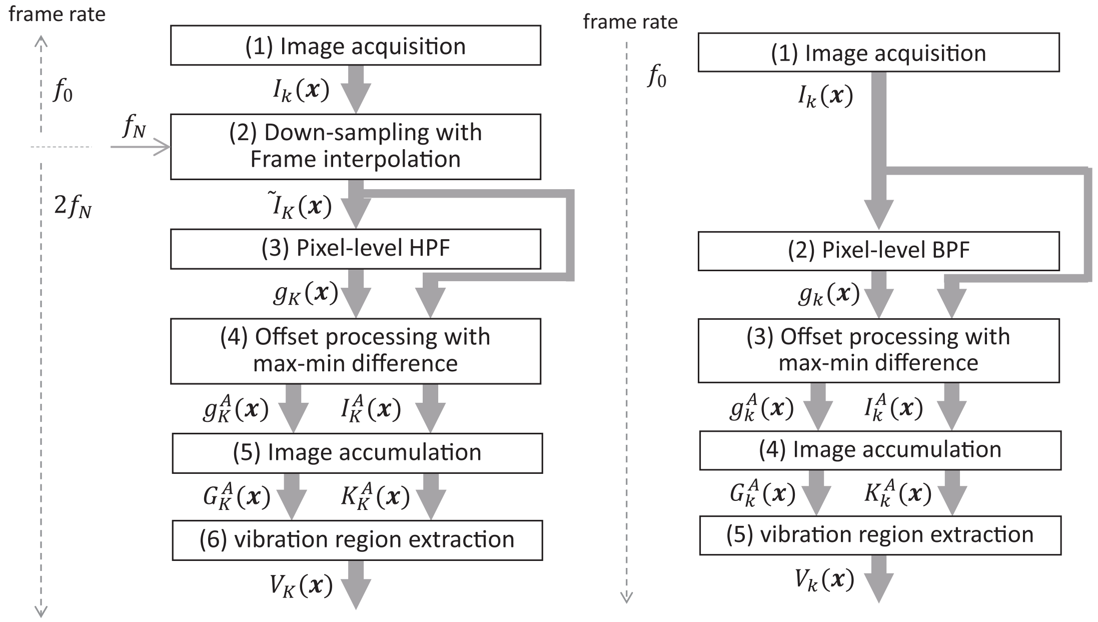
From an HFR camera operating at a fixed frame rate f_0, an input image at time $t = k\Delta t$ is captured as follows:

$$I_k(\boldsymbol{x}) = I(\boldsymbol{x}, k\Delta t), \tag{4.1}$$

where k, $\Delta t_0 = 1/f_0$, and $\boldsymbol{x} = (x, y)$ are the frame number, frame interval of the HFR camera, and coordinate value of pixels, respectively.

(2) Downsampling with frame interpolation

When the target's vibration frequency is $f_N(t)$ at time t, the virtual sampling time T_K



(a) our proposed method

(b) previous method

Figure 4.2: Flowcharts of vibration source tracking algorithms.

of the downsampled video $\tilde{I}_K(\boldsymbol{x}) = \tilde{I}(\boldsymbol{x}, T_K)$ is updated by adding twice the cycle time of the target vibration $T_N(T_{K-1}) = 1/f_N(T_{K-1})$ to the previous sampling time T_{K-1} as follows:

$$T_K = T_{K-1} + 2T_N(T_{K-1}), \tag{4.2}$$

Downsampled images at time T_K, $\tilde{I}_K(\boldsymbol{x}) = \tilde{I}(\boldsymbol{x}, T_K)$ are generated by executing linear frame interpolation between two input images $I_{k'}(\boldsymbol{x})$, $I_{k'+1}(\boldsymbol{x})$, which are sampled at the two nearest times around time T_K as follows:

$$\tilde{I}_K(\boldsymbol{x}) = \left((k'+1) - \frac{T_K}{\Delta t}\right) I_{k'}(\boldsymbol{x}) + \left(\frac{T_K}{\Delta t} - k'\right) I_{k'+1}(\boldsymbol{x}), \tag{4.3}$$

where k' is an integer that satisfies $k'\Delta t < T_K \leq (k'+1)\Delta t$.

(3) Pixel-level high-pass filters

By applying infinite impulse response (IIR) high-pass filters to time-varying brightness signals at all the pixels $\boldsymbol{x}$ in the downsampled images $\tilde{I}_K(\boldsymbol{x})$, the high-pass filtered images $g_K(\boldsymbol{x}) = g(\boldsymbol{x}, T_k)$ at time T_K are computed as follows:

$$g_K(\boldsymbol{x}) = \sum_{s=0}^{P-1} b_s \tilde{I}_{K-s}(\boldsymbol{x}) - \sum_{s=1}^{P-1} a_s g_{K-s}(\boldsymbol{x}), \tag{4.4}$$

where P is the filter order, and a_s, b_s are the tap coefficients; they determine the property of the high-pass filter. $\tilde{I}_K(\boldsymbol{x})$ and $g_K(\boldsymbol{x})$ are initially set to zero images. The symmetry frequency component has an aliasing effect on the downsampled images, concerning the Nyquist frequency f_N, enabling a high-pass filter with a pass frequency $(1 - a/2)f_N$ to equivalently function as a bandpass filter, whose center frequency and bandwidth are f_N and af_N, respectively.

(4) Offset processing

To remove the offset values in the downsampled images and their high-pass filtered images, the offset-difference images $I_K^A(\boldsymbol{x}) = I_K^A(\boldsymbol{x}, T_K)$ and $g_K^A(\boldsymbol{x}) = g_K^A(\boldsymbol{x}, T_K)$ at time T_K are computed by differentiating the maximum and minimum values of $\tilde{I}_K(\boldsymbol{x})$ and $g_K(\boldsymbol{x})$ for time $t = T_K - T_M$ to T_K as follows:

$$I_K^A(\boldsymbol{x}) = I_K^M(\boldsymbol{x}) - I_K^m(\boldsymbol{x}), \tag{4.5}$$

$$g_K^A(\boldsymbol{x}) = g_K^M(\boldsymbol{x}) - g_K^m(\boldsymbol{x}). \tag{4.6}$$

The maximums and minimum images are computed as follows:

$$I_K^M(\boldsymbol{x}) = \max_{T_K - \Delta T_M < t \leq T_K} \tilde{I}(\boldsymbol{x}, t), \quad I_K^m(\boldsymbol{x}) = \min_{T_K - \Delta T_M < t \leq T_K} \tilde{I}(\boldsymbol{x}, t), \tag{4.7}$$

$$g_K^M(\boldsymbol{x}) = \max_{T_K - \Delta T_M < t \leq T_K} g(\boldsymbol{x}, t), \quad g_K^m(\boldsymbol{x}) = \min_{T_K - \Delta T_M < t \leq T_K} g(\boldsymbol{x}, t), \tag{4.8}$$

where ΔT_M corresponds to M times of the vibration cycle time $\Delta T_N = 1/f_N$. ΔT_N is twice the frame interval of the downsampled images. The abovementioned process is executed for $2M$ frames, corresponding to the M cycle time.

(5) Image accumulation

The downsampled images $I_K^A(\boldsymbol{x})$ and their high-pass filtered images $g_K^A(\boldsymbol{x})$ after the offset processing are accumulated in a certain interval of ΔT_L as follows:

$$K_K(\boldsymbol{x}) = \int_{T_K-\Delta T_L}^{T_K} I_K^A(\boldsymbol{x}, t)dt, \tag{4.9}$$

$$G_K(\boldsymbol{x}) = \int_{T_K-\Delta T_L}^{T_K} g_K^A(\boldsymbol{x}, t)dt, \tag{4.10}$$

where ΔT_L is set to L times the cycle time ΔT_N.

(6) Vibration region extraction

The vibration region $V_K(\boldsymbol{x})$ is extracted by thresholding the ratio of the accumulated images of the downsampled images $K_K(\boldsymbol{x})$ to those of their high-pass filtered images $G_K(\boldsymbol{x})$, with a threshold θ_2 as follows:

$$V_K(\boldsymbol{x}) = \begin{cases} 1 & (G_K(\boldsymbol{x}) > \min(\theta_1, \theta_2 \cdot K_K(\boldsymbol{x})) \\ 0 & \text{(otherwise)} \end{cases}, \tag{4.11}$$

where pixel $\boldsymbol{x}$ is judged as an ambiguous pixel that should not be extracted when $G_K(\boldsymbol{x})$ is less than the threshold θ_1.

The difference of vibration source localization for HFR images at a fixed frame rate in [12] is that "(2) downsampling with frame interpolation" is added and the pixel-level bandpass filters in [12] are replaced with "(3) pixel-level high-pass filters"; the other sub-processes are similar to those in the previous method. Our algorithm can reduce its computational complexity and required memory size in vibration source localization, corresponding to the target's vibration frequency. It can accelerate the speed of pixel-level digital filters by introducing, pixel-level digital filters, downsampling with frame interpolation, in which the Nyquist frame rate of downsampled images is tracked to the target's vibration frequency using high-pass filters, whose filter order is half that of bandpass

Table 4.1: Computational complexities per pixel per unit time ($R = f_0/f_N$)

	our proposed method		
	ADD / SUB	MUL	COMP
(1) image acquisition	–	–	–
(2) down sampling	$2f_N$	$4f_N$	–
(3) digital filter	$4Pf_N$	$(4P+2)f_N$	–
(4) offset processing	$4f_N$	–	$16Mf_N$
(5) accumulation	$8Lf_N$	–	–
(6) vibration extraction	–	$2f_N$	$4f_N$
total	$(8L+4P+6)f_N$	$(4P+8)f_N$	$(16M+4)f_N$

	previous method		
	ADD / SUB	MUL	COMP
(1) image acquisition	–	–	–
(2) down sampling	–	–	–
(3) digital filter	$4PRf_N$	$(4P+1)Rf_N$	–
(4) offset processing	$2Rf_N$	–	$4MR^2f_N$
(5) accumulation	$2LR^2f_N$	–	–
(6) vibration extraction	–	Rf_N	$2Rf_N$
total	$(2LR^2+(4P+2)R)f_N$	$(4P+2)Rf_N$	$(4MR^2+2R)f_N$

filters.

Here, we will discuss vibration source localization with pixel-level digital filters when the translation speed of the target to be observed is not much larger than its vibration frequency. We assume that the number of cycle times when the time-varying brightness signal related to the target's vibration is observable at the same pixel, and $N_p = A \cdot f_N/v$ is sufficiently large for computing with digital filters; A, f_N, and v are the size, vibration frequency, and translation speed of the target to be observed, respectively.

4.4 The number of calculations and computational complexities

Next, we will evaluate the number of calculations for adding and subtracting, multiplication, and comparison operation in this proposed method if we track vibration source for target frequency $f_N(= f_0/R)$, which is $1/R$ of frame rate f_0. Table 4.4 shows the summary of the number of calculations which we need in 1 pixel per time unit. Here, we remove the execution time regarding (1) Image acquisition. Comparing the previous method, we also wrote the number of calculations with the previous method [12] is vibration source tracking for a fixed frame rate. In the case of the proposed method, the number of calculations in (2) Downsampling using frame interpolation takes 1 adding and subtracting and two multiplication every time we generate 1 frame of interpolated image.

The number of calculations of filter order P of (3) processing high pass filter need $2P$ adding and subtracting and $2P + 1$ multiplication. (4) Offset processing with max-min difference in M period of target frequency needs 2 adding and subtracting and $8M$ comparison operation. (5) Image accumulation in L period of target frequency needs $4L$ adding and subtracting. Finally, (6) Vibration source localization needs 1 multiplication and 2 comparison operation. When we calculate the number of calculations per unit time at 1 pixel is $2f_N$ times of that number. So, Total of those are number of adding and subtracting $D^{fi}_{add} = (8L + 4P + 6)f_N$, number of multiplication $D^{fi}_{mul} = (4P + 8)f_N$, and number of comparison operation $D^{fi}_{comp} = (16M+4)f_N$. On the while, in the conventional vibration source tracking method, the downsampling process of (2) is not used for comparison, but the frame of (3) is used instead of a high-pass filter with filter order P for the input image at frame rate f_0. A bandpass filter with filter order $2P$ with level performance is directly applied, and the processing from (4) to (6) is the same as the proposed method. For each frame of the input image, adding and subtracting is performed $4P$, and multiplication is perfomed $4P + 1$ for the $2P$ order bandpass filter processing in (3). For the M period of the vibration in (4), offset processing using maximum/minimum value operation has $2P$ adding and subtracting and $4Mf_0/f_N = 4MR$ multiplication. It is necessary to perform adding and subtracting $2Lf_0/f_M = 4LR$ times in the integrated image calculation for L cycles of the vibration in (5), one multiplication and two comparison operations in vibration area extraction in (6), The number of calculations per pixel in the unit time is $f_0 = Rf_N$

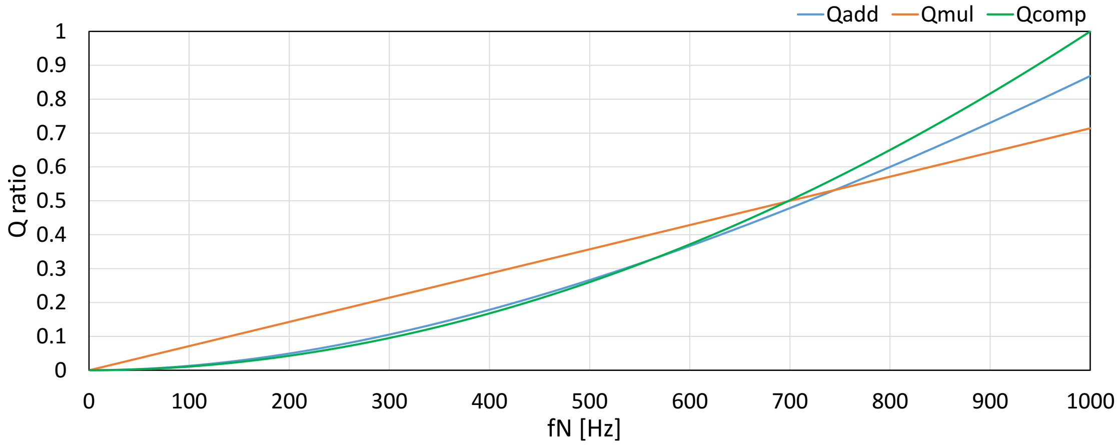


Figure 4.3: Relationship between computational complexities and target vibration frequency ($P = 3, M = 3, L = 6$)

times these, and if expressed in the form of R and f_N, as shown in Table 4.4. Total numbers of calculations for the previous method are the number of adding and subtracting $D^{pr}_{add} = (2LR^2 + (4P+2)R)f_N$, number of multiplication $D^{pr}_{mul} = (4P+2)Rf_N$, and number of comparison operation $D^{pr}_{comp} = (4MR^2 + 2R)f_N$. The ratio of adding and subtracting Q_{add}, the ratio of multiplication Q_{mul}l and the ratio of multiplication Q_{comp}, per time unit between the proposed method and the previous method are given as follows:

$$Q_{add} = \frac{D^{fi}_{add}}{D^{pr}_{add}} = \frac{4L + 2P + 3}{R((2P+1) + LR)}, \quad Q_{mul} = \frac{D^{fi}_{mul}}{D^{pr}_{mul}} = \frac{2P+4}{R(2P+1)},$$
$$Q_{comp} = \frac{D^{fi}_{comp}}{D^{pr}_{comp}} = \frac{8M+2}{R(2MR+1)} \tag{4.12}$$

When the target frequency f_N is satisfied with $R \geq 2$ which means target frequency is lower than Nyquist frequency $f_0/2$, each of Q_{add}, Q_{mul} and Q_{comp} are $(4L+2P+3)/(4L+4P+2)$, $(P+2)/(2P+1)$, 1 and they are max value when $R = 2$. These equations indicate that the proposed method can be faster than the previous method if the target frequency is less than Nyquist frequency because all of ratio are less than 1 when filter order P is larger than 1 without considering the value of L and M.

For instance, Figure 4.3 shows the relationship between the number of calculations ratio in Q_{add}, Q_{mul}, Q_{comp}, and target frequency when parameters set as the filter order $P = 3$, period of offset processing with max-min difference $M = 3$, the period of image accumulation $L = 6$ for input framerate $f_0 = 2000$ fps. These parameters are following the experiment's setting in section 4.7; the ratio in Q_{add}, Q_{mul}, and Q_{comp} can be small and small when target frequency is much smaller than input framerate. For example, when we set target frequency $f_N = 200$ Hz as $R = 10$.

The ratio of adding and subtracting Q_{add}, the ratio of multiplication Q_{mul}l and the ratio of multiplication Q_{comp} are respectively 0.049, 0.143, 0.043. As a result, we can see the proposed algorithm is faster than the previous algorithm because we can reduce the number of adding and subtracting, number of multiplication and number of comparison operation at 1 pixel per time unit.

4.5 Filter performance

When we want to speed up the function of vibration source localization, we need to realize much less order of pixel-level digital filter than one of the previous. At the same time, we maintain filter performance as well as previous. Much research has the purpose that it can reduce the noise for the acoustic signal and so on, while it maintains phase characteristics and restrains the ripple of the signal. However, our proposed algorithm regarding vibration source localization has the purpose of whether it can detect the pixel, which has a distinct frequency component for image intensity. That's why it is not such a severe problem to ripple in passband and stopband in a filter and non-linear phase response with IIR filter because these image intensity's phase periodically changes at different timing pixel by pixel in general. On the other hand, setting narrow transition bandwidth that we can separate passband with stopband is essential for filter

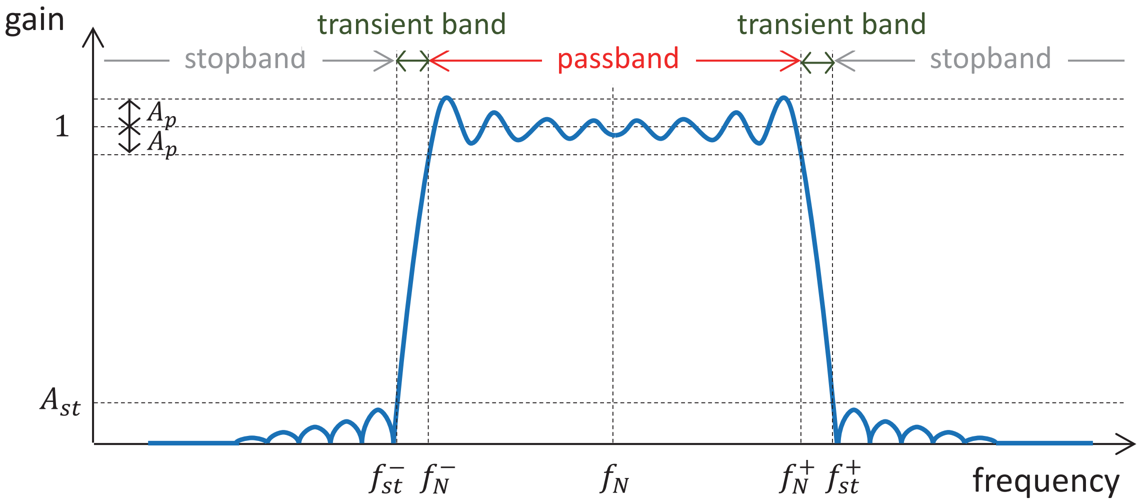


Figure 4.4: Transient band in digital filter.

performance. We need to localize vibration source at pixel level from the point of view that we can restrain misdetection for frequency component except for passband around target frequency f_N. After that, we decide to consist of a pixel-level digital filter for the elliptic filter. We can design narrow transition bandwidth between passband and stopband without compromising the ripple of passband and stopband.

As Figure 4.4 shows, we will evaluate filter performance by calculating transition width $\Delta f = (f_{st}^+ - f_{st}^-) - af_N$ after calculating stopband $f < f_{st}^-, f > f_{st}^+$ ($f_N^- = (1 - a/2)f_N,\ f_N^+ = (1 + a/2)f_N,\ a = 0.1$) when we design elliptic filter which pass band $f_N^- < f < f_N^+$ around target frequency $f_N = 200$ Hz and we set the parameters are maximum passband ripple $A_p = 1$ dB and stopband gain $A_{st} = 40$ dB, These parameters are almost same with experiment's parameters which we set in section 4.7.

Figure 4.5 shows the gain characteristic of (a) case of using ecliptic high pass filter whose order $P = 2, 3, 5, 10$ in proposed method and (b) case of using ecliptic bandpass filter whose order is $2P$ in the previous method for input framerate $f_0 = 2000$ fps. It indicates the gain characteristic of frequency, which includes over Nyquist frequency (200 Hz) in downsampling image at 400 fps in (a) without considering double image error

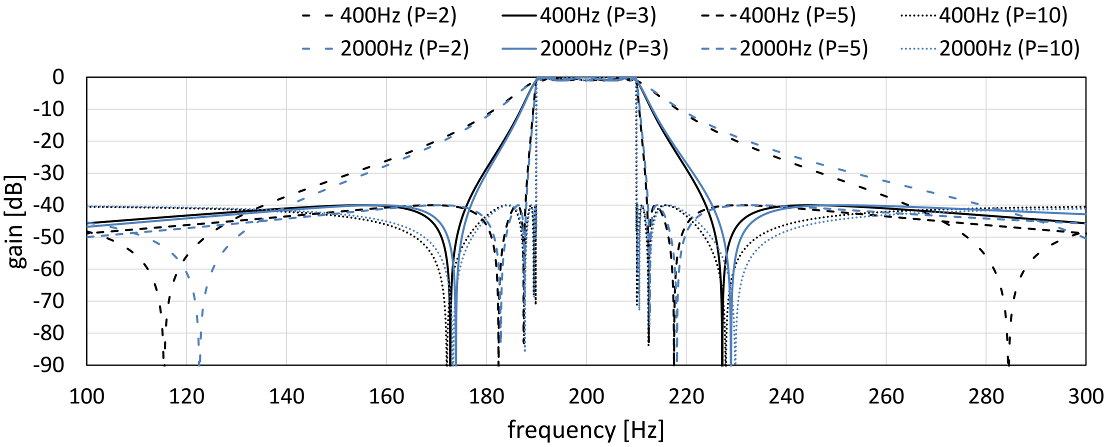


Figure 4.5: Gain properties of elliptic filters (P = 2, 3, 5, and 10).

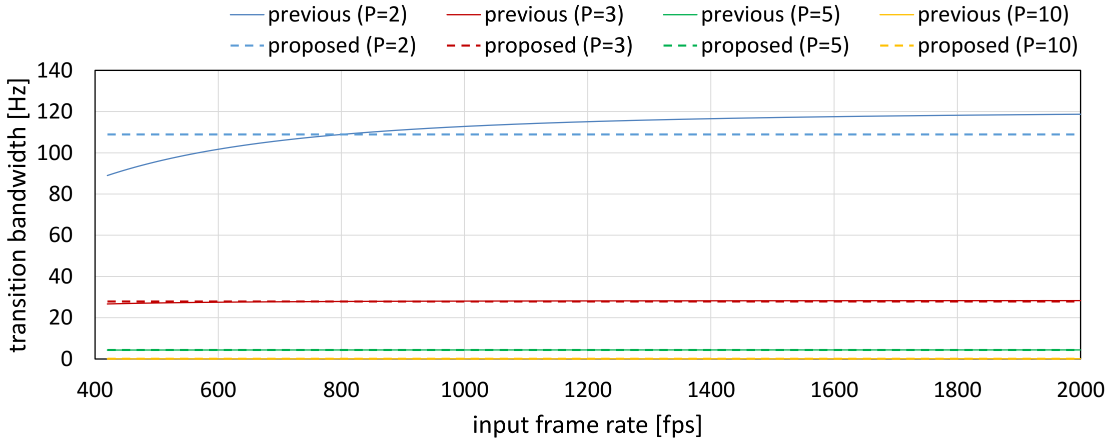


Figure 4.6: Relationship between input frame rate f_0 and transition bandwidth Δf

and effect because of downsampling with frame interpolation. Figure 4.6 indicates the relationship between framerate f_0 = and transition width Δf when we change framerate from f_0 = 420 fps to f_0 = 2000 fps. In the case of filter order P = 3, transition with of (a) is constantly 27.9 Hz regardless of input framerate, that of (b) in input framerate f_0 = 500, 1000, 1500, 2000 fps is 27.2, 28.0, 28.2, 28.3 Hz. So the results of (a) are almost the same with (b).

Next Figure 4.7 shows the relationship between filter order $P(= 1, \cdots, 10)$ and transition width. It indicates transition width of (a) and (b) when we set stopband gain A_{st} 40, 60, 80, 100 dB. If filter order P is larger and larger and input framerate f_0 = 2000 fps,

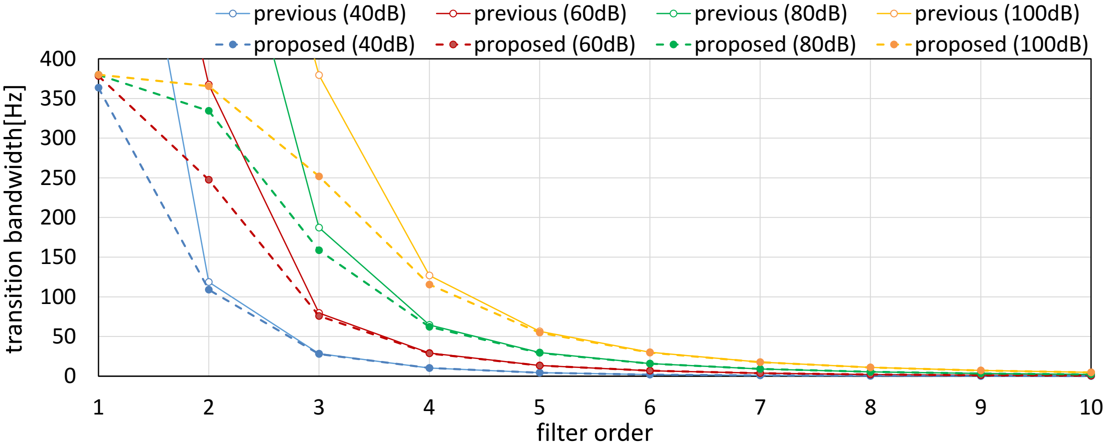


Figure 4.7: Relationship between filter order P and transition bandwidth Δf.

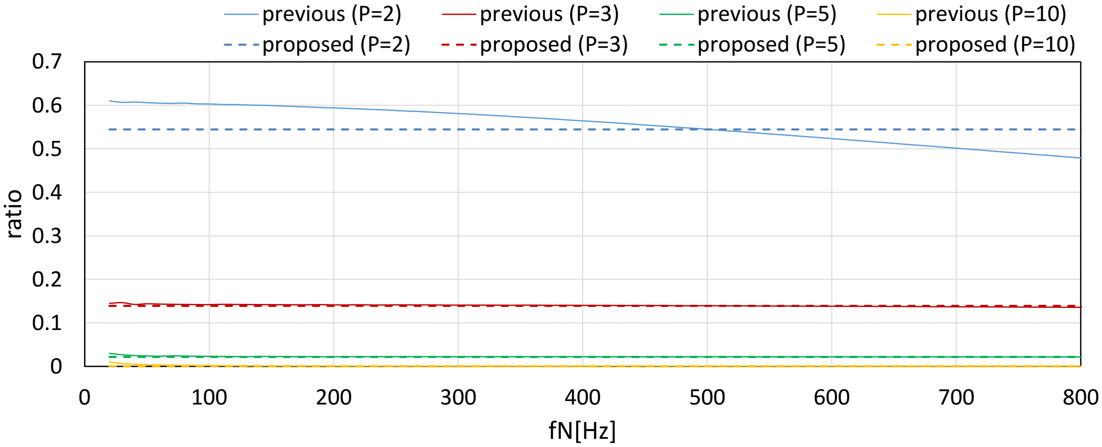


Figure 4.8: Relationship between target frequency f_N and the relative ratio of transition bandwidth to target frequency $\Delta f/f_N$

transition width of both (a) and (b) is smaller and smaller, and that of (a) is nearly corresponding with (b). For example, the transition width in $A_{st} = 40$ dB are corresponding with

(a) 108.9 Hz, (b) 118.7 Hz in $P = 2$, (a) 27.9 Hz, (b) 28.3 Hz in $P = 3$, (a) 4.4 Hz, (b) 4.4 Hz in $P = 5$ and (a) 0.1 Hz, (b) 0.1 Hz in $P = 10$. If a restriction of stopband gain is severe, the filter order needs large order to realize designing a desirable filter, which is narrow transition width from these graphs.

Figure 4.8 indicates the relationship between target frequency f_N and the relative ratio of transition width $\Delta f/f_N$ when we set the parameter of target frequency f_N from 20 Hz to 800 Hz in input framerate $f_0 = 2000$ fps. When the filter order P is 2, the relative ratio of (a) is a little bit different from (b) regarding target frequency. On the other hand, when the filter order P is larger than 3, the relative ratio of (a) is almost the same with (b). For example, when the filter order P is 3, a relative ratio of (a) is 0.139 regardless of target frequency and relative ratio of (b) in the target frequency $f_0 = 20$, 80, 200, 800 Hz is 0.145, 0.143, 0.142, 0.136. So the relative ratio of (a) is also almost the same with (b) in this case. As a result, when we design the digital filter which is the center of target frequency, we can see downsampling by frame interpolation in (a) are not almost effective for the filter performance regardless of the target frequency. That's why we demonstrate the algorithm which is vibration source tracking with pixel-level digital filter can speed up while the filter characteristic that we need to track vibration source maintains. This is because that the transition width especially is not so different from the case that we design the digital filter whose order is the same with the previous method for high framerate input images. It is regarding the proposed method in which we calculate pixel-level digital filter for the downsampling image as well as the previous method.

4.6 Execution time

Next, we evaluated the execution time after we implemented the proposed algorithm on the PC. We used a PC with an HP Z440 Workstation (Intel Xeon E5-1650v4 processer @ (3.60 GHz CPU, 6cores, 15MB, 2.4 GHz), Intel C612 chipset, 32GB RAM, and Windows 7 Enterprise 64-bit OS. Moreover, we use OpenMP [95] which makes multi-thread for parallel processing to speed up when we implement the algorithm. We evaluated the execution time for the proposed method, when we set the parameters as follow: vibration frequency $f_N = 200$ Hz, period of offset processing with max-min difference $M = 3$ and a period of image accumulation $L = 3$ as well as the parameters we set for offline tracking

Table 4.2: Execution times and image sizes (unit [s]).

			(1) image acquisition	(2) down sampling	(3) digital filter	(4) offset processing	(5) accumulation	(6) vibration region extraction	total
256 × 256	proposed	1 thr	—	0.046	0.391	0.346	0.423	0.019	1.226
		10 thr	—	0.009	0.055	0.055	0.076	0.004	0.199
	previous (500 fps)	1 thr	—	—	3.894	0.748	1.238	0.028	5.909
		10 thr	—	—	0.347	0.107	0.181	0.006	0.640
	previous (1000 fps)	1 thr	—	—	7.801	2.646	5.654	0.060	16.162
		10 thr	—	—	0.672	0.382	0.710	0.013	1.777
	previous (2000 fps)	1 thr	—	—	15.836	13.994	25.179	0.128	55.136
		10 thr	—	—	1.370	1.773	3.284	0.029	6.456
512 × 512	proposed	1 thr	—	0.206	2.480	1.671	2.115	0.089	6.561
		10 thr	—	0.033	0.367	0.253	0.402	0.016	1.072
	previous (500 fps)	1 thr	—	—	15.893	3.759	6.581	0.121	26.354
		10 thr	—	—	1.476	0.559	0.832	0.023	2.889
	previous (1000 fps)	1 thr	—	—	30.329	13.641	24.167	0.231	68.368
		10 thr	—	—	2.894	1.783	3.646	0.050	8.373
	previous (2000 fps)	1 thr	—	—	61.814	58.702	179.196	0.494	300.206
		10 thr	—	—	5.887	9.055	25.483	0.105	40.528
1024 × 1024	proposed	1 thr	—	0.791	9.130	8.956	13.800	0.338	33.014
		10 thr	—	0.123	1.859	1.461	1.910	0.067	5.419
	previous (500 fps)	1 thr	—	—	61.484	20.398	31.271	0.456	113.609
		10 thr	—	—	5.869	2.664	3.660	0.089	12.282
	previous (1000 fps)	1 thr	—	—	122.571	67.995	108.238	0.914	299.718
		10 thr	—	—	11.621	7.863	14.787	0.177	34.448
	previous (2000 fps)	1 thr	—	—	175.137	199.587	524.962	1.403	901.089
		10 thr	—	—	23.447	36.787	93.866	0.370	154.470
2048 × 2048	proposed	1 thr	—	3.030	35.461	34.278	53.336	1.314	127.419
		10 thr	—	0.475	8.035	5.356	7.476	0.270	21.613
	previous (500 fps)	1 thr	—	—	175.717	58.774	91.418	1.354	327.263
		10 thr	—	—	22.867	10.038	14.159	0.335	47.400
	previous (1000 fps)	1 thr	—	—	462.547	258.162	411.075	3.495	1135.279
		10 thr	—	—	45.719	30.040	56.749	0.661	133.169
	previous (2000 fps)	1 thr	—	—	694.673	791.742	2067.934	5.477	3559.826
		10 thr	—	—	91.720	143.438	361.117	1.361	597.636

experiment for multi-copter as we will explain about it in chapter 4.7.

Table 4.2 shows the breakdown of execution time for input image as 256×256, 512×512, 1024×1024 and 2048×2048 in input framerate $f_0 = 2000$ fps in a second's time. Table 4.2 indicates the intermediate value of execution time when we measure the execution time at 10 times trials with same processing. It also indicates breakdown of execution time regarding case of fixed framerate ($f_0 = 500$, 1000, 2000 fps) to compare fixed framerate. The execution time in both of thread 1 and thread 10 are increasing if the image size is increasing. In other words, we can clearly see the tendency when execution time in previous method is longer than that in proposed method especially in case of that input framerate is so high. For example, if it is 512×512 image at input framerate $f_0 = 500$, 1000, 2000 fps, execution time in previous method is longer than proposed

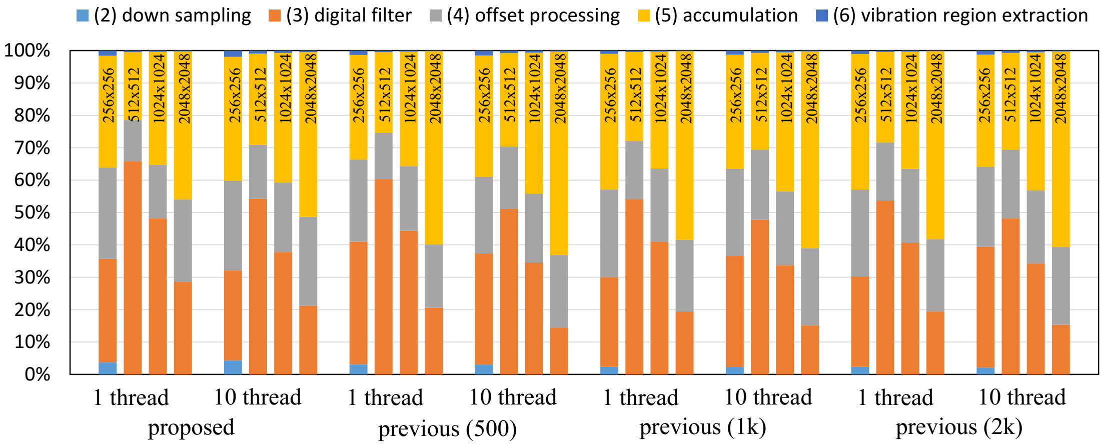


Figure 4.9: Breakdown percentages of execution times.

method as follow : 4.0, 10.4, 45.8 times in thread 1 and 2.7, 7.8, 37.8 times in thread 10. Both method's execution time of proposed and previous in thread 10 are shorter than that in thread 1 regardless of image size as its ratio is about execution time in previous method is longer than proposed method as follow : 4.0, 10.4, 45.8 times in thread 1 and 2.7, 7.8, 37.8 times in thread 10. Both method's execution time of proposed and previous in thread 10 are shorter than that in thread 1 regardless of image size as its ratio is about $\frac{1}{6}$–$\frac{1}{6}$ by using multi thread processing. Figure 4.9 indicates occupancy of each processing for execution time. And both method's occupancy is mainly consisted of (3) digital filter, (4) offset processing and (5) Image accumulation in more than 90% regardless of image sizes and number of thread. In this case, we can see (2) down sampling which we introduced in proposed method is not interfere with other part of processing.

Next, Figure A.2 shows the relationship between execution time for input image in a second's time in both method and image size when we make number of pixel M change from 20 to 1000 by every aproximately10 pixels in case of image size is $M \times M$ pixel. The parameters show as follow: input framerate f_0 = 2000 fps, number of thread 10 and filter order P =3. Execution time for input image in a second's time in Figure A.2 is within 1 second. In other words, maximum size of image which we can execute

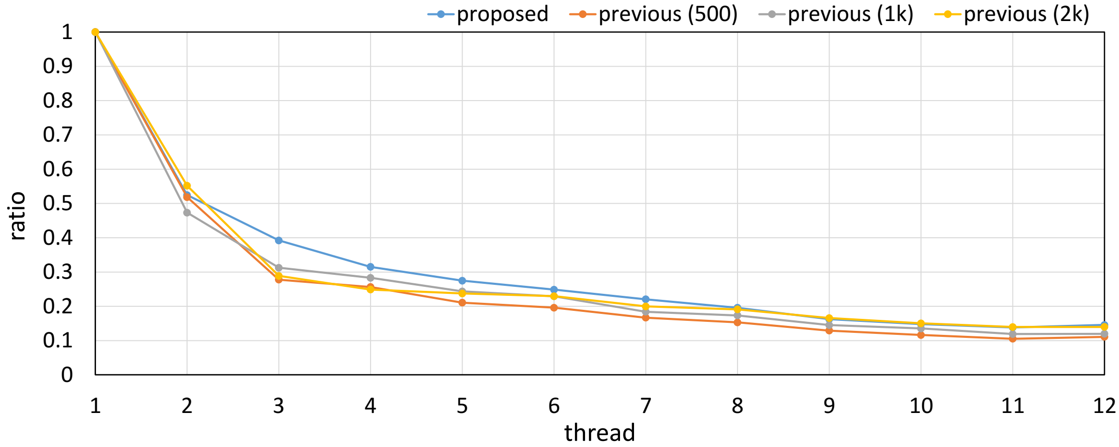


Figure 4.10: Relationship between execution times and image sizes.

on real time using thread 10 is 550×550 pixels in proposed method while that in previous method at input framerate f_0 = 500,1000,2000 fps is 340×340,240×240,130×130 pixels respectively. So the number of pixels in previous method we can process on real time is 0.40, 0.20, 0.06 times as much as that in proposed method. As you can see these results, it is possible for proposed algorithm to realize vibration source localization for much higher space resolution than previous algorithm.

Figure 4.11 indicates relationship between execution time and number of thread when we set number of threads as 1–12 in proposed algorithm (P=3) for 512×512 input image in a second's time at input framerate f_0 = 2000 fps. Here, we show relative change of execution time in previous method and proposed method for number of threads as the execution time in thread 1 is standard. The execution times in thread 1 are 6.561 s in proposed algorithm and 23.314, 60.761, 268.990 s in previous algorithm (f_0 = 500, 1000, 2000 fps). When number of threads is larger than 10, execution regarding number of threads is almost same in both algorithm while execution time in previous and proposed algorithm is decreasing when number of threads is increasing. The execution times in thread 10 are 0.971 s in proposed algorithm and 2.458, 7.232, 37.555 s in previous algorithm (f_0 = 500, 1000, 2000 fps). Comparing with thread 1, execution time in previous algorithm is 0.15 times and execution times in previous algorithm(f_0 = 500, 1000,

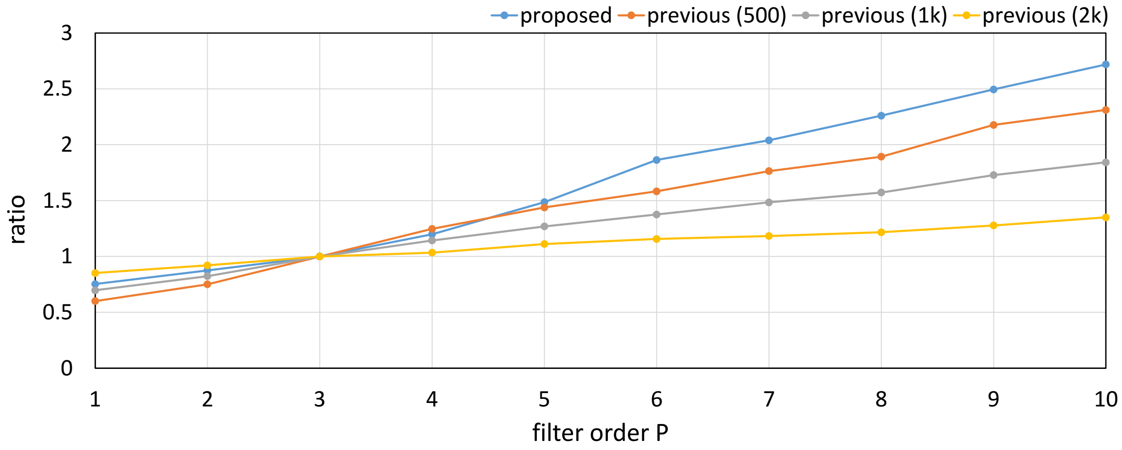


Figure 4.11: Relationship between execution times and number of threads.

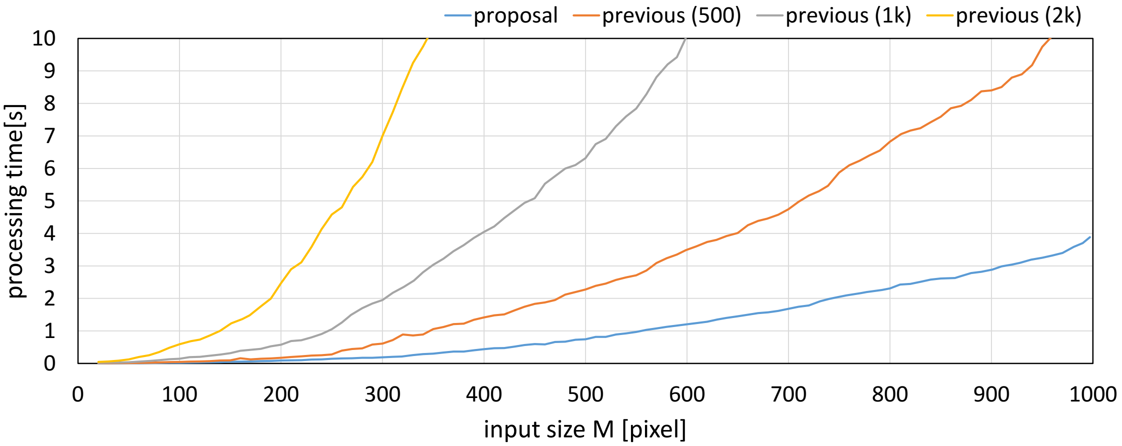


Figure 4.12: Relationship between execution times and input frame rate.

2000 fps) are respectively 0.12, 0.14, 0.15 times, so the execution time in thread 10 speed up around 7 times faster than thread 1. These execution times are depending on number of PC processor cores which we evaluated, and it is corresponding to implementation of vibration source tracking algorithm with multi thread by synchronizing Windows OS with other standard software for processor Intel Xeon E5-1650v4 which provide 12 processing thread with 6 physical cores.

Figure 4.12 indicates execution time in proposed and previous method (P =3) for 512×512 input image in a second's time in thread 10 when input framerate f_0 changes between 400 and 2000 fps. The execution time in proposed algorithm is always 0.971 s

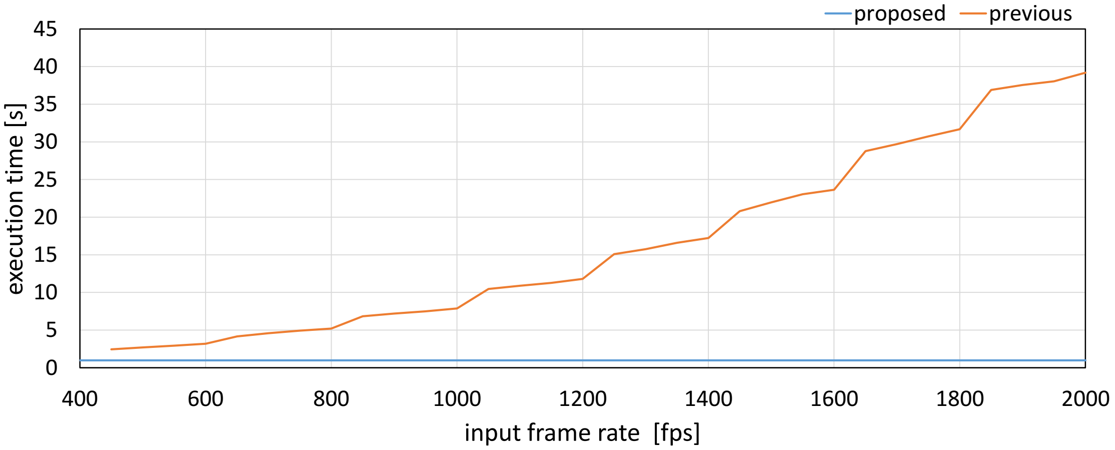
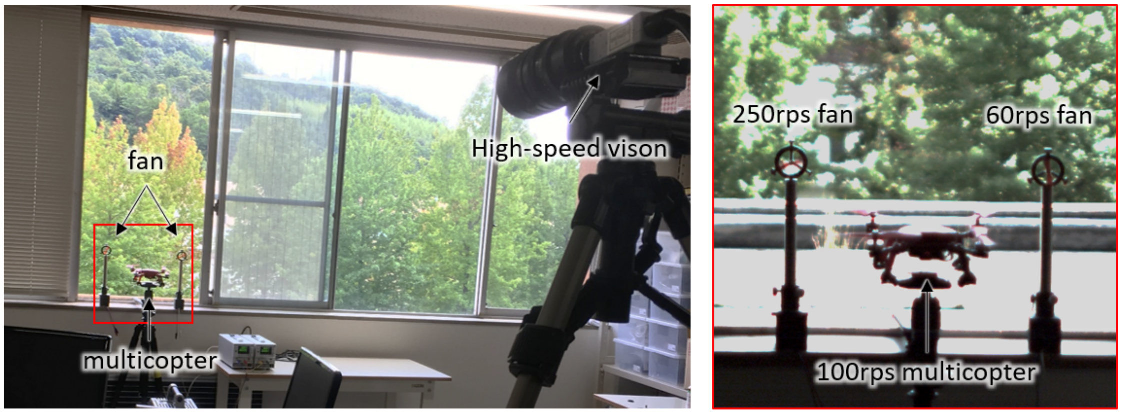


Figure 4.13: Relationship between execution times and filter orders.

by introducing down sampling processing with frame interpolation regardless of input framerate. On the other hand, the execution time in previous algorithm (f_0 = 500, 1000, 2000 fps) is increasing with increasing input framerate as we explained before and indicates 2.707, 7.866, 41.150 s respectively. For example, prosed method is over 40 times faster than previous one in case of f_0 = 2000 fps. The proposed method dramatically can expect to speed up for previous method when input framerate f_0 is much higher than target frequency f_N = 200Hz because computational complexity is decided by dependence of the target frequency f_N = 200Hz.

Last, Figure 4.13 indicates the relationship between execution time and filter order P in both previous and proposed method for 512×512 input image in a second's time at input framerate f_0 = 2000 fps, when we set filter order P = 1–10. In Figure 4.13, we can show relative change of execution time in previous method and proposed method for filter order as the execution time in filter order P =3 is standard. The execution times in proposed algorithm and in previous algorithm (f_0 = 500, 1000, 2000 fps) are increasing as filter order P is increasing. In addition, execution times in filter order P =10 is 2.7 times in proposed method and 2.3, 1.8, 1.3 times longer than that in previous algorithm in P =3. Moreover, the effect of execution time in previous algorithm for filter order changing tends to be small when input framerate f_0 is higher than target frequency f_N.



(a) experimental environment

(b) multicopter and fan

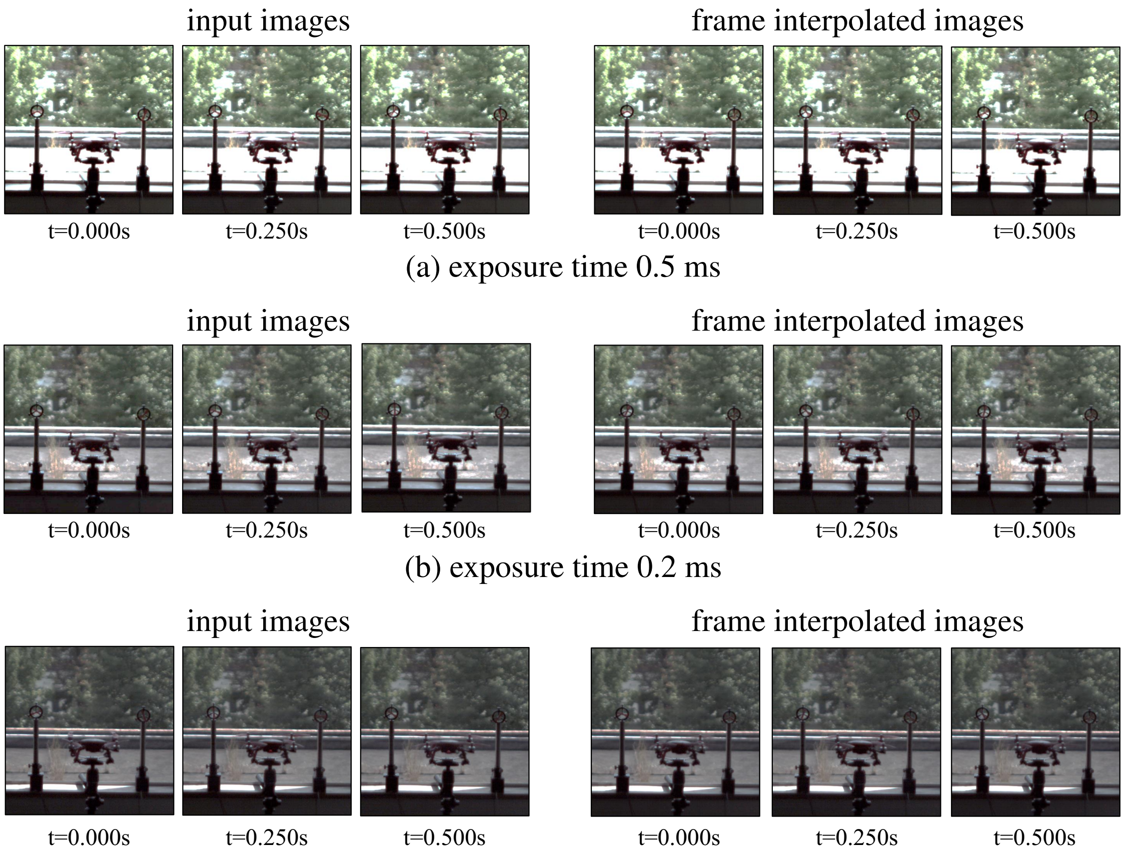
Figure 4.14: Experimental environment when video-shooting a fixed multi-copter.

This is because that computational complexity of (4) offset processing and (5) image accumulation which need to process the number of periods for vibration frequency as number of frame in a vibration period is increasing with increasing input framerate is increasing regarding that filter order *P* for (2) digital filter in previous method can be fixed value.

4.7 Experiments for a multi-copter

4.7.1 Video shooting a fixed multi-copter with various exposure time

Firstly, we checked wheter we could extract region in which multi-copter's propeller rotate as vibrated region in proposed algorithm as well as previous vibration source localization. Considering about that there are double image errors which occurs in case that exposure time is longer than the period of framerate in linear interpolated image after recording 512×512 image of 8 bit color for a fixed multi-copter and fans at 2000 fps, when we set exposure time as 0.5, 0.2, 0.1 m. We selected to use multi-copter (RC EYE One Extreme by CEI Conrad company) which is size of object is 225×225×80 mm and have 4 two-blade propellers whose each size is 138 mm for evaluation. Figure 4.14 indicates



(a) exposure time 0.5 ms

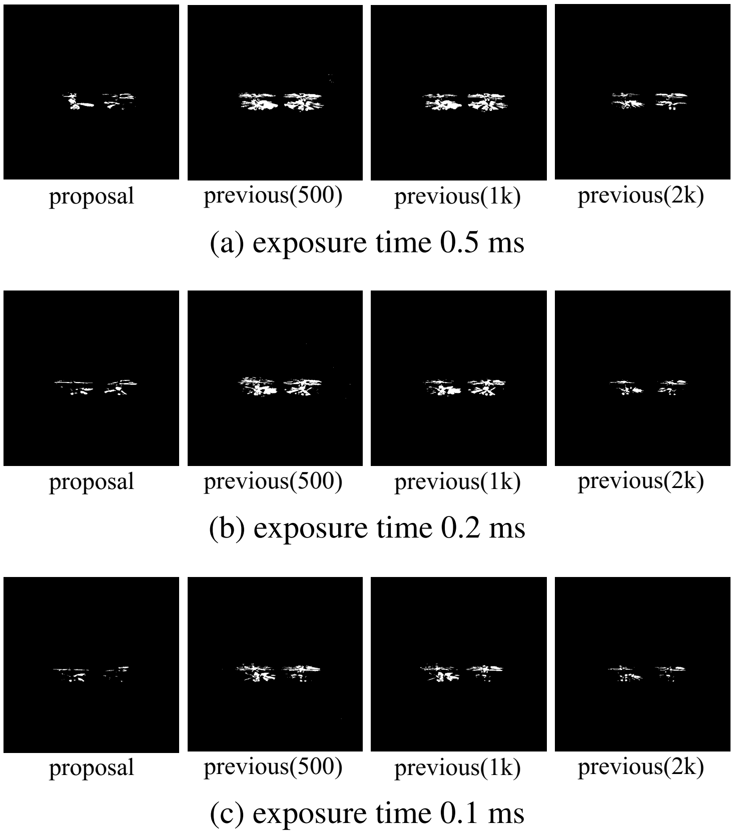
(b) exposure time 0.2 ms

(c) exposure time 0.1 m

Figure 4.15: Input images when video-shooting a fixed multi-copter.

these experimental environment. We set the distance as 4.5 m from camera to multi-copter and a pixel is corresponding to 1.0 mm for multi-copter's position and 512×512 image is corresponding to 512×512 mm. In addition, we fixed the body of multi-copter at an angle in order to monitor all of motion of 4 propellers. And we also fixed 2 two-blade fans whose rotation speed is set as 250, 60 rotation per second respectively next to multi-copter as we also can see background of trees on sun light. We make multi-copter's propeller rotating at 100 rps in all setting in this section. We set parameters in proposed algorithm as follow: vibration frequency f_N = 200 Hz, filter order P=3, period of offset processing with max-min difference M =3, period of image accumulation L =6.

Figure 4.15 indicates input frames and frame interpolated images' group in proposed method when we set exposure time (a) 0.5 ms, (b) 0.2 ms, (c) 0.1 ms, frame inter-

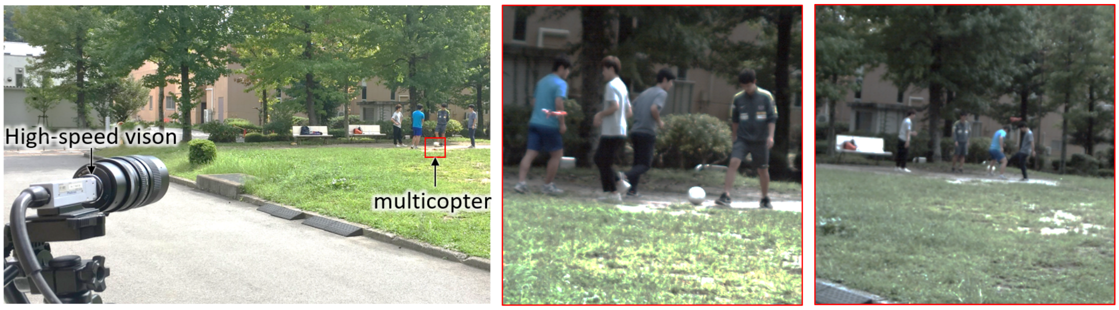


(a) exposure time 0.5 ms

(b) exposure time 0.2 ms

(c) exposure time 0.1 ms

Figure 4.16: Extracted vibration features when video-shooting a fixed multi-copter.



(a) experimental environment (b) scene 1 (b) scene 2

Figure 4.17: Experimental environment for flying multi-copter.

polated images can be double image error when exposure time is longer especially while motion blur in propeller's region can be reduced and input image turns dark all over the image.

Figure 4.16 shows vibration region by using proposed method as we compare vibration region by implementing previous algorithm (f_0 = 500, 1000, 2000 fps) in the case of exposure time (a) 0.5 ms, (b) 0.2 ms, (c) 0.1 ms. We can detect areas in which there are multi-copter's propeller rotating at 100 rotation per second as vibration region while we don't detect areas in which there are fixed 2 two-blade fans whose rotation speed is set as 250, 60 rps as vibration region in both methods. So this tendency is not depending on exposure time. Furthermore, in terms of the number of detected pixels, when input framerate $f_0 = 2000$ is in the first case by 100 and 500 fps for previous method in Figure 4.16. This is because that there are some pixels whose difference are small between max and min, and number of sampling is not enough to extract actual intensity change in case of low framerate. So the number of pixels detected in equation (4.11) by calculating 30 frames, 15 frames and 8 frames ($f_0 = 500, 1000, 2000$ fps) regarding maximum and minimum brightness for 15 ms ($f_N = 200$ Hz, $M = 3$), which is corresponding to equation (4.7). On the other hand, number of detected pixels in proposed algorithm in equation (4.11) is decreasing because Nyquist frequency is corresponding to target frequency and number of pixel which we can detect maximum and minimum brightness as its phase is almost inversely at 180 degrees is increasing and number of pixel whose difference between max and min is much larger than previous method regardless of input framerate $f_0 = 400$ fps. As a result of these considerations, we can realize vibration source tracking in proposed algorithm performs as well as that in previous algorithm regardless of double image errors when the exposure time is small.

4.7.2 Video shooting a flying multi-copter

Next, we will show results of vibration source tracking for 512×512 image of 8 bit color for a flying multi-copter in outdoor at 2000 fps as background is walking humans and trees and so on when we apply the proposed algorithm. We used multi-copter which is same one in chapter 4.7.1. Figure 4.21 indicates the experimental environment. Here, we can monitor brightness change in enough period at the same pixel in this experiment because of parameter $N_p = 35.4$ as followed two-blade propeller's size A is 138 mm, target frequency $f_N = 200$ Hz, maximum speed $v = 0.78$ m/s. We recorded the experimental images at 2 patterns as (1) case with large magnification and (2) case with wide angle. We set the distance as 24.0 m from camera to flying multi-copter and 512×512 image is corresponding to 3072×3072 mm in (1) case and 7680×7680 mm in (2) case for multi-copter's position and a pixel is corresponding to 6.0 mm in (1) case and 15.0 mm in (2) case. In addition, we used the parameters are same with previous indoor experiment. Figure 4.18 indicates time sequential images of (a) input images with tracked positions, (b) accumulated images of input $K_K(\boldsymbol{x})$, (c) pixel-wise filtered images $G_K(\boldsymbol{x})$, (d) vibration features extracted by our proposed algorithm $V_K(\boldsymbol{x})$, (e) vibration regions extracted by previous algorithm (f_0 = 2000 fps) and (f) ROI input images for tracked regions whose gravity position is center of this image, when video-shooting a flying drone with large magnification. The rectangular region in (a) is corresponding to magnified 50×50 ROI image of (f). We can extract the vibration region which is corresponding to region of multi-copter's propeller rotating in proposed algorithm as well as previous algorithm even though multi-copter moved around areas in full image. In addition, we can specify the vibration source region which is same as position of flying multi-copter because there is always multi-copter in center of ROI image of (f).

Figure 4.22 indicates time sequential image of (a) input images with tracked positions, (b) accumulated images of input, (c) pixel-wise filtered images, (d) vibration features extracted by our proposed algorithm and (e) previous algorithm (f_0 = 2000 fps) and (e) magnified images for tracked regions when video-shooting a flying drone with a wider view angle. Even though multi-coper's image is very low resolution's image which corresponds to 30×30 image as Figure 4.22 (f) shows it, comparing with zoom image which corresponds to about 50×50 pixels'region, we can realize pixel level vibration region extraction for region of multi-copter's propeller in proposed algorithm as well as that in previous algorithm regardless of effect for noise which is moving humans and something around natural tree in background. We can track multi-copter as vibration region because there is multi-copter in center of ROI image in almost all time although there are some cases that the position of multi-copter is a little bit different from center of the image due to some occlusion which the propeller sometimes overlaps multi-copter itself.

Figure 4.23 shows (a) time sequential data of gravity position and number of pixels regarding vibration region for 10 seconds and (b) xy trajectory of extracted vibration region when a flying drone was video-shot with a wider view angle. In order to compare with the results of extracted region in previous method, it also is shown in Figure 4.23 Moreover trajectory in Figure 4.23 (b) is plotted on input image in time t =0 in Figure 4.22 (a). The gravity position of vibration region in proposed method nearly corresponds to that in previous method as it is same with up, down, left and right of movement of multi-copter even though number of pixel in extracted vibration region change from 0 to 54 significantly when multi-copter moved in front of walking human and trees as background. Furthermore, we also can realize robust vibration source tracking with pixel level digital filter for flying multi-copter though there are some occlusions which the propeller sometimes overlaps multi-copter itself and some cases that the gravity position of multi-copter is a little bit different from center of ROI image.

4.7.3 Real-time multi-copter detection

We show the experimental results of real-time vibration source detection when a multi-copter moved against a background with trees and moving persons. The multi-copter flew to the right and left directions above many people with quick arm and leg movements. The 512×512 input images were captured at 2000 fps with 0.5 ms exposure. We used multi-copter which is same one in chapter 4.7.1. The flapping frequency of each propeller varied by approximately 95 rps. Figure 4.21 shows the experimental environment. The multi-copter flew at a distance of approximately 17.4 m from the camera. The measurement area was 5.24×5.24 m for 512×512 pixels at a distance of 17.4 m from the camera, where one pixel corresponds to 10.2 mm. The target vibration frequency was set to $f_N = 190$ Hz and the pass bandwidth in the digital filters was 20 Hz so that the pixel-level digital filters can pass the frequency components of time-varying brightness signals in the range of 180–200 Hz. The parameters were set to $M = 3$, $L = 6$, and $P = 3$. Our algorithm was executed in real time with multithread processing of 10 threads.

Figure 4.22 (a)–(c) shows the input images, the accumulated images of input and high-pass filtered images. The images were taken at intervals of 1 s for $t = 0$–5 s. Figure 4.22 (d)–(e) shows the vibration pixels extracted by our algorithm and the previous algorithm in [12] when $f_0 = 2000$ fps. Figure 4.22 (f) shows the magnified images of 80×80 pixels around the vibration pixels extracted by our algorithm, corresponding to the red-line windows illustrated in (a). When the multi-copter flew in the right and left directions against the cluttered background with trees and moving persons, our algorithm extracted certain pixels around the propellers of the multi-copter in real time. We can confirm that the regions of interest illustrated in Figure 4.22 (f) included the multi-copter at all the times, whereas their space resolution was too low for precise appearance-based recognition.

Figure 4.23 illustrates (a) the graphs that show changes in the x- and y-coordinate values of the averaged positions of the extracted pixels and the number of extracted pixels for 5 s; and (b) the xy trajectory for 5 s was plotted over the input image at $t = 0$. Corresponding to the left and right movements of the flying multi-copter, the xy trajectory of the averaged positions of the extracted pixels were robust without any disturbance from the cluttered background of the outdoor location.

4.8 Concluding remarks

In this chapter, we demonstrated vibration source tracking for object's frequency is 200 Hz with input images at 2000 fps in proposed algorithm is over 10 times faster than that in previous method after we propose a fast algorithm which reduce computational complexity per unit time by controlling sampling period for digital filter with downsampled HFR images whose frame rates were virtually adjusted to twice the target vibration frequency with frame interpolation, while locking in vibration frequency for tracking object, we check the number of calculations and filter performance and we evaluate execution time on PC with CPU regarding vibration source tracking for high framerate video images with pixel level digital filter. We confirmed that the execution time of our algorithm was 1/10 or less than that of the previous algorithm when localizing a vibration source using HFR images captured at 2000 fps. The effectiveness of this porposed algorithm was verified by the experimental results for HFR offline video analysis and real-time detection for multi-copter, in which a multi-copter with propellers rotating at 95 rps that flew in a cluttered outdoor location was simultaneously localized using 512×512 images at 2000 fps. Furthermore, in this section, we focused on speeding up for pixel level vibration source localization which can extract frequency information in image pixel by pixel and we discussed with the case that frequency of multi-copter's propeller is obviously higher than frequency of background and the other object's motion. However, robust recognition method which integrated dynamics characteristic for acoustic frequency level with space appearance characteristic can be meaningful if we consider about the case that there are various kinds of dynamic background and multiple similar objects in a point of view in actual scene. Moreover, vibration source localization for downsampled image which controlled locking in the sampling period for peak frequency can also be valid by detecting peak vibration frequency based on intensity change for acoustic signal and image because the number of rotations of propeller and so on is not constant for condition of multi-copter flying and depending on various kinds of that. Vibration source tracking algorithm we proposed in this section will be able to process much more pixels than now and we will improve this system which can track multi objects at the same time by utilizing that we can set variable framerate corresponding to various vibration frequency with software after implementing the function of detecting vibration frequency automatically in future work. Finally, we will improve our vision-based vibration source localization system with real-time feedback control of its frame rate so that it can robustly localize and track a flying multi-copter when the rotation frequencies of its propellers are unknown or vary over time..

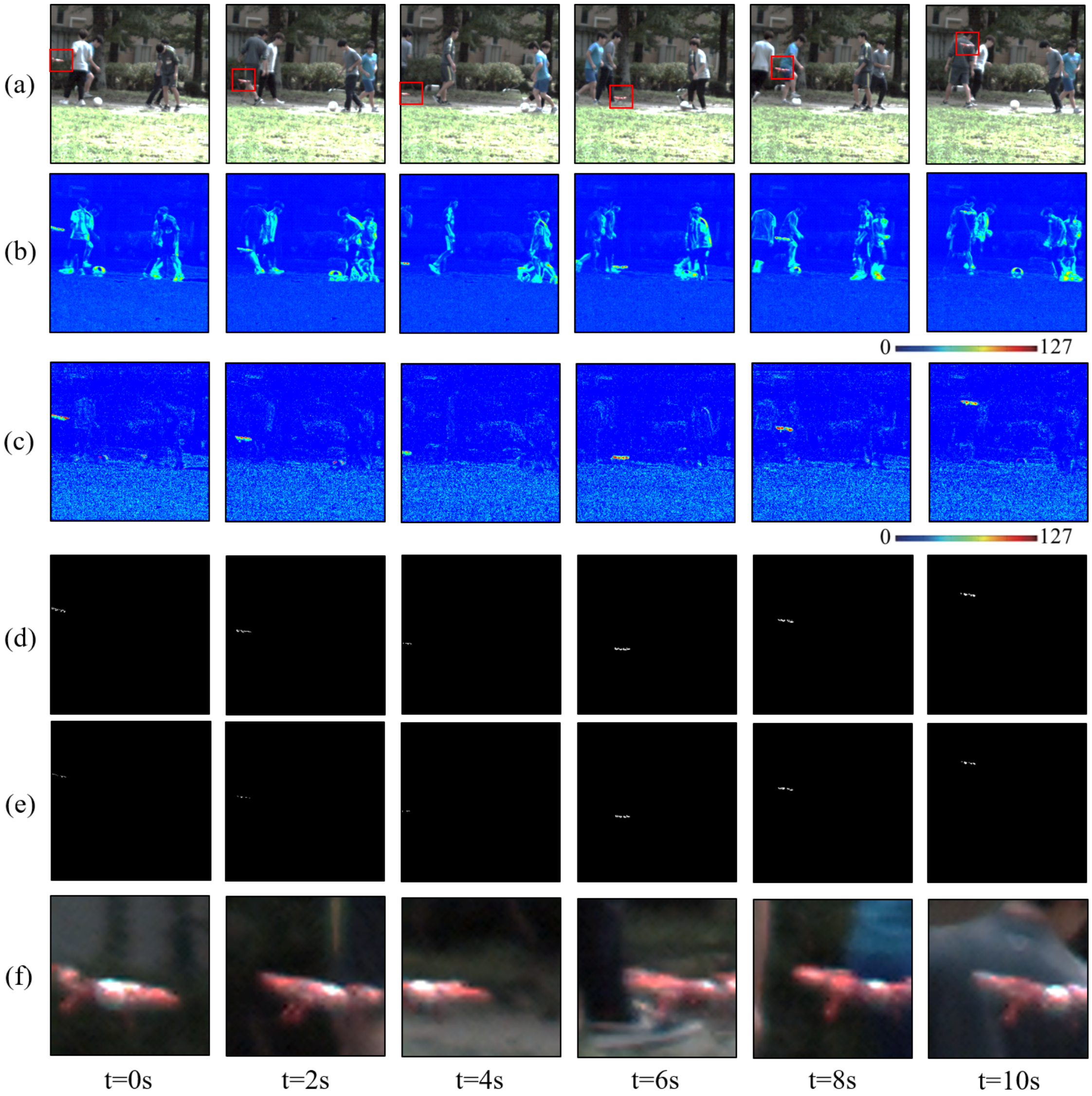


Figure 4.18: (a) Input images with tracked positions, accumulated images of (b) input and (c) pixel-wise filtered images, vibration features extracted by (d) our proposed algorithm and (e) previous algorithm (f_0 = 2000 fps), and (e) magnified images for tracked regions when video-shooting a flying drone with large magnification.

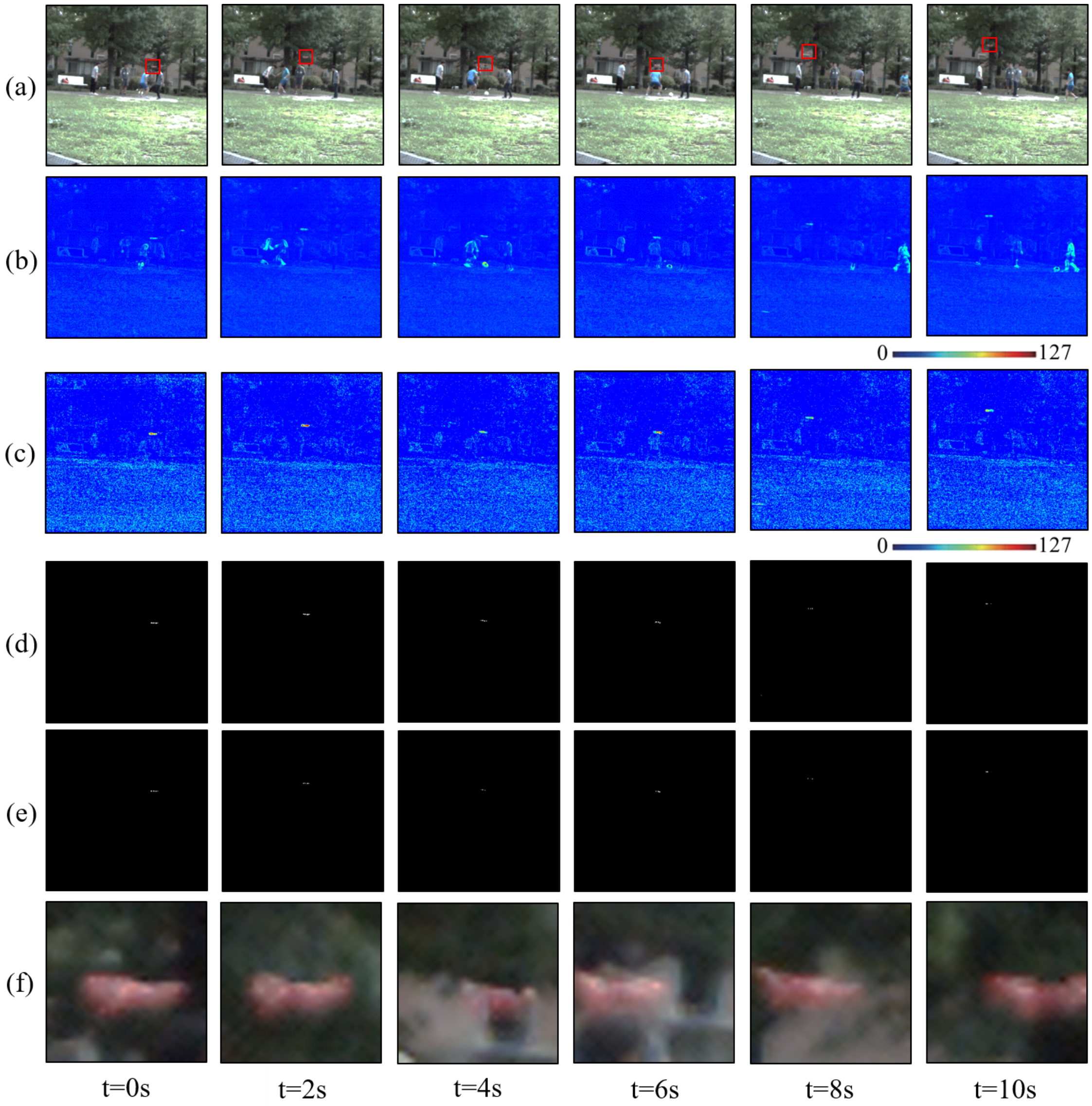
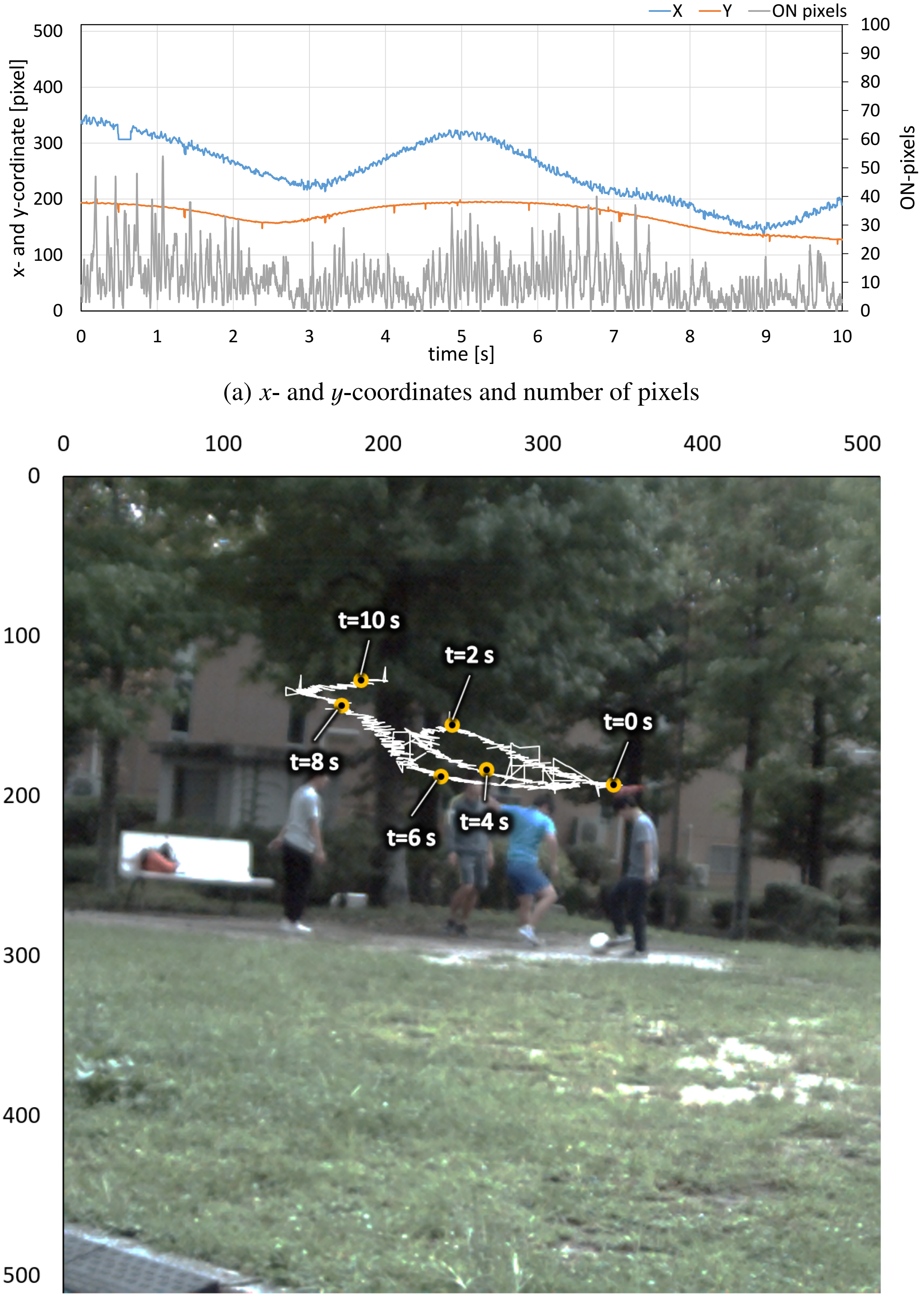


Figure 4.19: (a) Input images with tracked positions, accumulated images of (b) input and (c) pixel-wise filtered images, vibration features extracted by (d) our proposed algorithm and (e) previous algorithm (f_0 = 2000 fps), and (e) magnified images for tracked regions when video-shooting a flying drone with a wider view angle.



(a) x- and y-coordinates and number of pixels

(b) xy trajectory

Figure 4.20: xy trajectory of extracted vibration region when a flying drone was video-shot with a wider view angle.

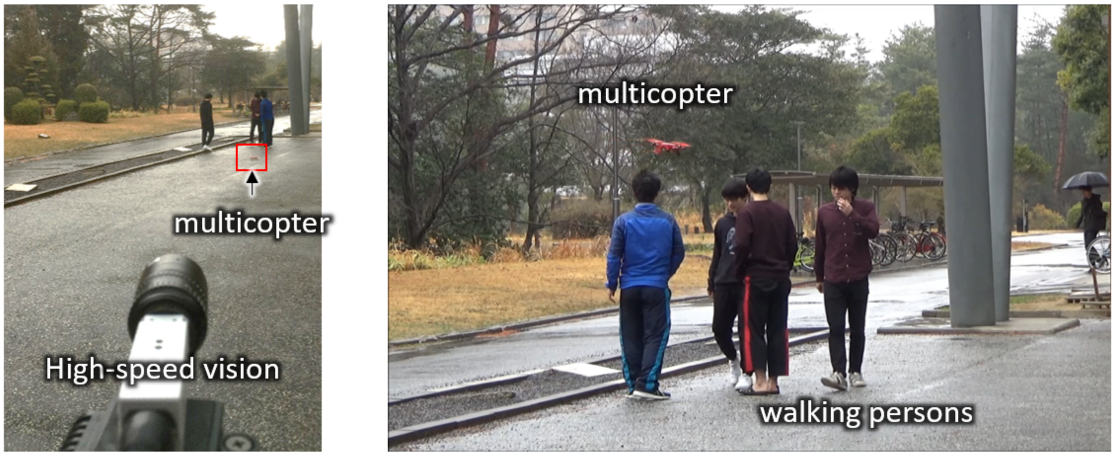


Figure 4.21: Experimental environment for flying multi-copter.

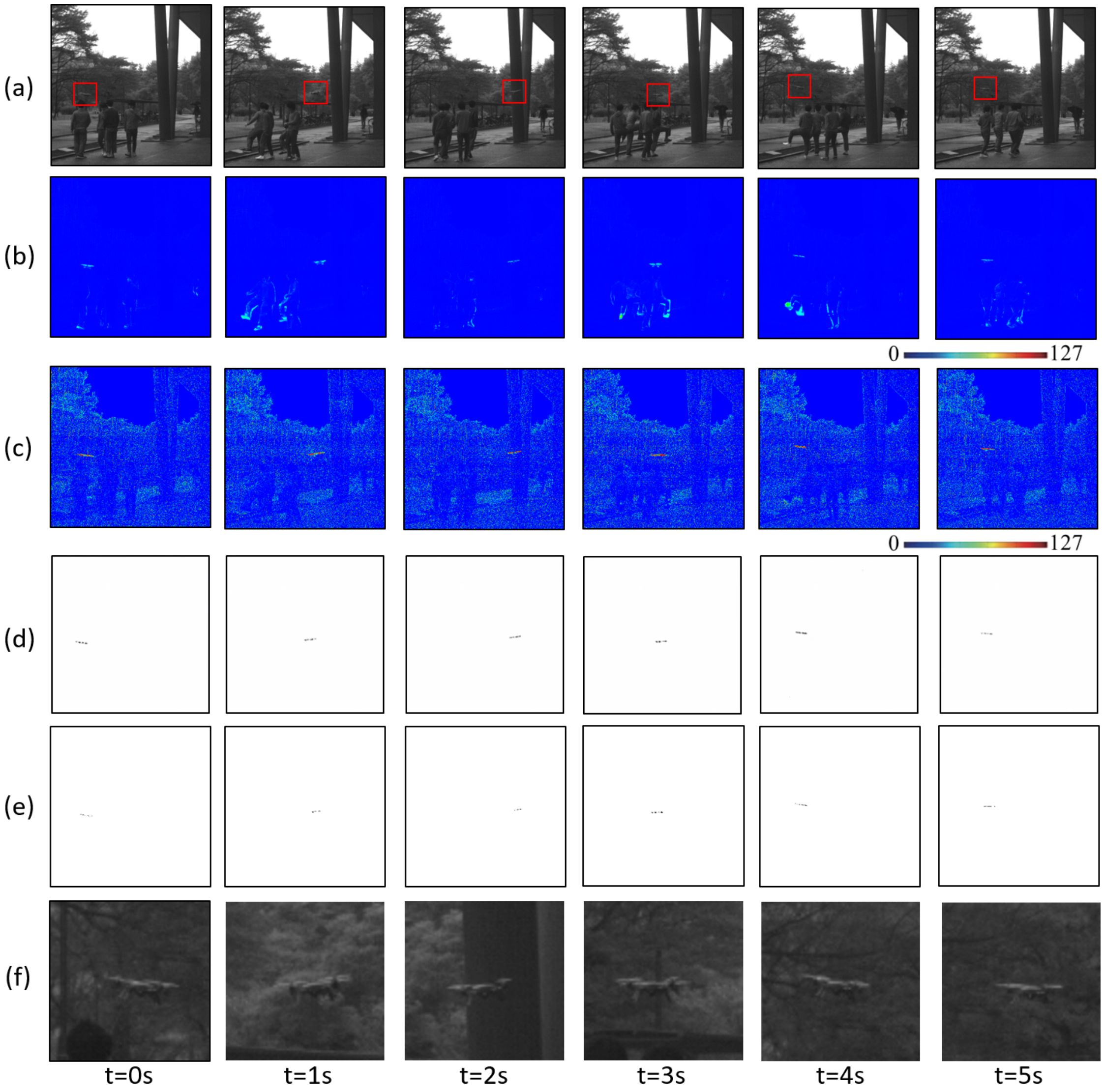
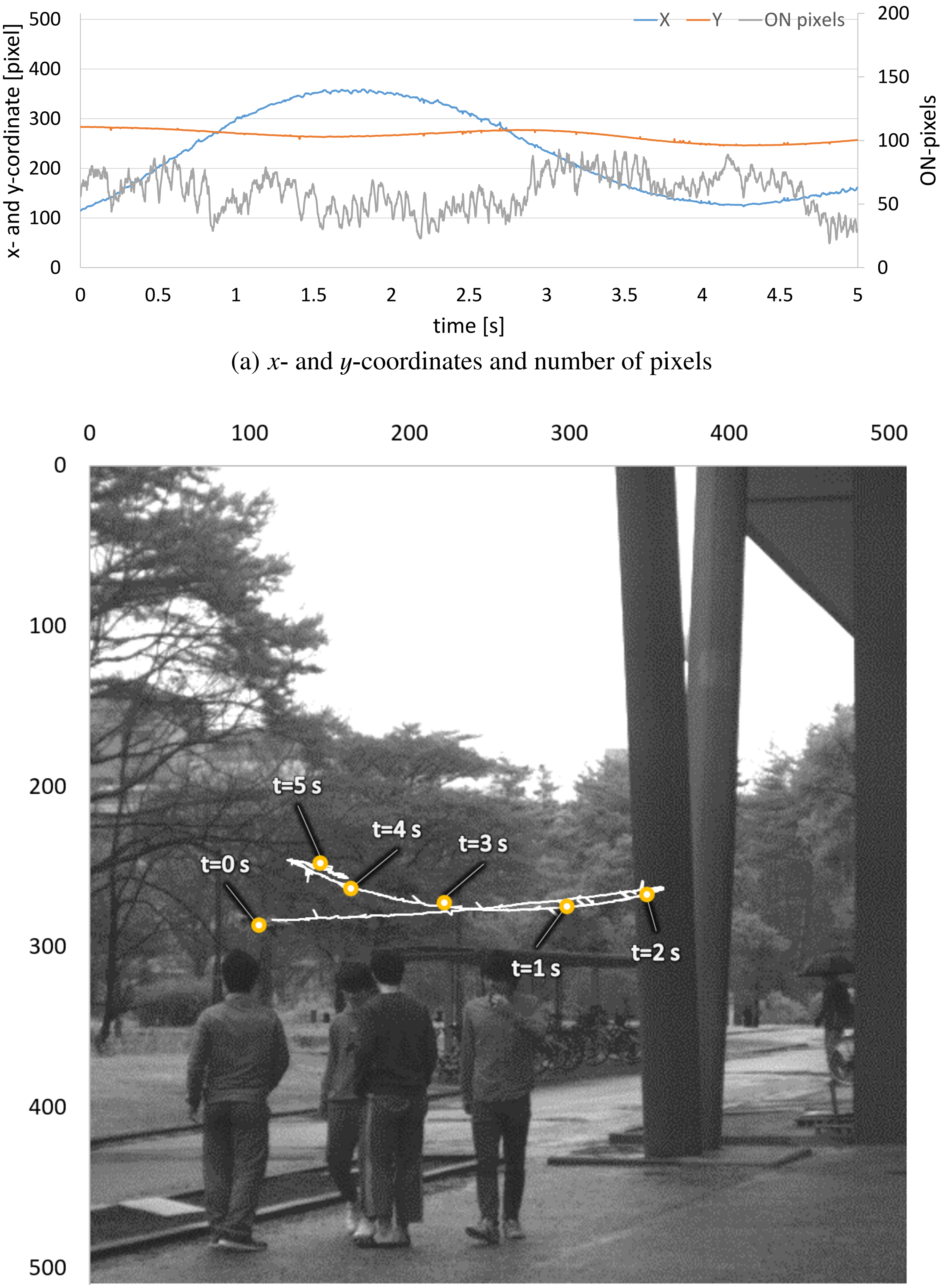


Figure 4.22: (a) Input images; accumulated images of (b) input; (c) high-pass filtered images; vibration pixels extracted by (d) our algorithm; and (e) previous algorithm (f_0 = 2000 fps); (f) magnified images.



(a) x- and y-coordinates and number of pixels

(b) xy trajectory

Figure 4.23: The xy trajectory of extracted vibration region when the video of the flying drone was shot with a wider angle.

Chapter 5

HFR-video-based honeybee activity sensing

5.1 Introduction

Presently, the honeybee industry is of huge benefit to agricultural businesses by pollinating the flowers of crops, vegetables, and fruits as well as by producing and storing honey; the honeybee is recognized as one of the most efficient insect pollinators that can transport and store pollen [96]. Colony collapse disorder (CCD) is the phenomenon where the majority of worker bees in a colony disappear rapidly, leaving behind the queen. The number of disappearances of beehives has increased drastically in the 2000s [97]; CCD has become a serious social problem that causes significant economic losses to agricultural business because honeybees pollinate many agricultural crops. Whereas the causes of CCD are not perfectly clarified yet, various possible causes such as plagues and viruses, malnutrition, pesticides and miticides, beekeeping practices, electromagnetic radiation, and genetically modified crops have been reported [98–100]. To obtain useful knowledge for efficient beehive management without CCD outbreak, it is necessary to monitor colony activity to elucidate the reason for this syndrome [101].

Honeybee colonies monitoring had been attempted already more than 100 years ago; Gates [102] have reported the temperature data of a beehive collected manually every hour over several days. The rapid progress of sensor technology in recent years has resulted in the use of various types of electronic sensors such as (1) weight, (2) temperature, humidity, and gas, (3) sound and vibration, and (4) forager traffic [103] to conduct the continuous monitoring of honeybee colonies.

Beehive weight is related with colony growth and daily colony food demand as well as colony component such as adults, brood, and food reserves. To provide information on various issues such as weather effects [104], colony growth and consumption [105], and hive abandonment [106], continuous weighing with a precise electronic balance has been conducted. Temperature, gas concentration, and humidity are related with the metabolic processes of the colony level; calculations from temperature and O_2 consumption can be used to estimate the energy produced by a colony. Monitoring with these modalities has been conducted to observe colony temperature control against decreasing ambient temperature [107], low O_2 level due to reduced metabolic rate when water / energy conservation, and longevity [108], daily cycle in O_2 linked to forager activity [109], and time-varying humidity near the brood in a beehive [110].

Vibration and sound observed in a hive or its surface involve biological information that bees generate wing-flapping vibrations at many frequencies in the range of 10 to 1000 Hz to communicate such as the waggle dance [111–114]. The wing-flapping vibration of a single bee was measured with laser vibrometry and analyzed with STFT to examine it in the time and frequency domains [115]. By embedding an accelerometer in the wall of hives, Bencsik et al. [116] analyzed its vibration data with principal components analysis to examine its vibration frequencies to the noise in swarming events robustly. By conducting 270-hour monitoring, several swarming events were observed with increased sound intensity and decreased temperature and humidity in the hives [117]. One-year sound intensity data with several frequency bands was analyzed to quantify how the sound in healthy hives differs from that in chalkbrood infected hives [118].

Forager activity [119, 120], indicating how many older bees were gathering food, is important to monitor honeybee colony health as well as food availability / demand, and colony age structure. Hive entrance counters such as an electrical counter with a balance arm [121], photoelectric sensors [122, 123], and infrared light sensors [124] have been developed. Danka et al. [125] measured forager traffics during blueberry pollination, and faced a miscounting problem with photoelectric sensors because the bees tended to cluster around the sensors; they improved the measurement accuracy by validating forager flights with a visual count. Several attempts at forager traffic monitoring have been reported by attaching RFID chips to bees [126, 127]; they can collect crucial information such as life expectancy time and foraging time because bees tagged with chips are identified individually. The chips are small and weigh a few milligrams, however, it was too laborious and expensive to attach them to numerous bees, and the hive entrance should be modified to set up RFID receivers. Several lidar-based systems [128–130] have been reported for the large-area detection of honeybees. By detecting their wing-flapping in the audio frequency range using mechanical laser scanning, they can measure the 3-D positions of flying honeybees at a distance of hundreds of meters. However, they cannot keep track of a specific flying honeybee to monitor its flying activity for a certain interval.

Activity analysis of honeybees inside beehives using automatic bee tracking and behavior labeling has been reported in [131, 132]. Many systems and studies have analyzed the flight activities of honeybees outside beehives with capacitance-sensor-based monitoring [133], wingbeat frequency analysis [134], radio-frequency identification (RFID) chips for honeybee counting [135], and computer-vision-based monitroing such as honeybee detection with background subtraction [136, 137], automated bee-counting system [138, 139], and three-dimensional bee tracking [140]; they used standard videos recorded at dozens of fps, in which the wing-flapping of bees were unobservable, and because appearances of small honeybees in images became unclear without large magnification, analyzable scenes were limited to small areas around the hive entrance. Most of the researchers have handled honeybee monitoring in an exceedingly limited narrow area around a beehive. Especially in vision-based analysis, there is a tradeoff between the measurement area and detection accuracy, because the size of a honeybee is a dozen of millimeters; honeybees cannot be distinguished manually by their appearances in images when the measurement area such as 1×1 m or more is much larger than their size.

As we introduced appearance-based tracking methods in subsection 2.2, the limitation of these methods in tracking honeybees flying in a natural scene is the appearances of honeybees change and become ambiguous in the low resolution images. By detecting temporal periodic changes in brightness signals, HFR video analysis has been applied to vibration distribution measurement such as structural vibration analysis [141–143] and vocal fold vibration [144–146]. If honeybees with wings flapping at hundreds of Hertz could be observed in an HFR video, activity sensing of honeybees flying within an area that was a few meters in diameter around a beehive, which was much larger than the size of a honeybee, could be realized without attaching sensors by using pixel-level digital signal processing; this will help to quantify the colony health before and after CCD or other serious incidents. Pixel-level digital signal processing was used for honeybee localization in [147] to extract pixels around honeybees. When a honeybee is not hovering in the same place, however, it is difficult to precisely obtain its wing-flapping frequency as its flying activity. This is because one or no wave is observed in the image intensities at pixels around fast-flying honeybees.

Therefore, we propose a concept of honeybee activity sensing that can precisely obtain the wing-flapping frequencies of fast-flying honeybees as well as their positions by executing two-step short-time Fourier transforms (STFTs) for an HFR video, which is an extension of the honeybee localization algorithm in [147]. Pixel-level STFTs are executed at all the pixels of an HFR video as the 1st-step process for honeybee localization in a similar manner as in [147], and pixel-level STFTs of the image intensities around honeybees are re-executed at all the pixels of the tracked region-of-interest (ROI) images as the extended 2nd-step process to precisely obtain their wing-flapping frequencies. This is because honeybees are virtually located at the centers in the tracked ROI images, and multiple waves can be observed for precise frequency response computation.

5.2 Concept

Western honeybees are often used in beekeeping all over the world; the sizes of adult worker bees are 12–14 mm, and their wing-flapping frequencies when flying are 170–270 Hz in the audio frequency range [148]. It is difficult for a video-rate camera operating at dozens of fps to detect such a small flying honeybee under natural complex background conditions. Regarding an HFR camera as a collection of photo sensors, this study proposes a quantification method for the flying activities of multiple honeybees in images with pixel-level digital filters. Our method is an extension of the vision-based vibration source localization approach [12], in which a pixel-level digital filter had been used for localization whereas no pixel-level frequency response analysis had been considered. Figure 5.1 shows the concept of HFR-video-based activity sensing that this study presents.

Fast-flying honeybees in an HFR video are localized with pixel-level digital signal processing when their appearances are poor in the low resolution and blurred images [147]. There are still many problems to precisely obtain their wing-flapping frequencies as their flying activities. Here, the maximum flying speed of a western honeybee was reported as 25 km/h in [149], and because this movement is so large, one or no wave is observed at the same pixel. When a 12-mm-size honeybee is flying at 25 km/h, the dwell time in a single pixel, where its wings are detectable, is 1.7 ms in the worst case; it is smaller than one cycle time of its wing-flapping. Figure 5.2 illustrates the problems in pixel-level digital signal processing for fast-flying honeybees. The temporal sequences of image intensities at the pixels to be analyzed are largely dependent on background scenes. Excluding a short time for one-or-no-wave observation of their flapping wings, it is difficult to use pixel-level digital signal processing to obtain precise TFRs in image intensities at pixels around fast-flying honeybees. A distinct peak in the frequency response might not be detected for one-or-no-wave observation, as illustrated in Figure 5.2(a). In addition, the latency effect in pixel-level digital signal processing [12, 13] enlarges when a fast-flying honeybee is being observed; several pixels are incorrectly detected as ghost vibration regions even when the honeybee is not currently at these pixels owing to the latency effect in the pixel-level STFTs, as illustrated in Figure 5.2(b). A 12-mm-size honeybee at a flying speed of 25 km/h moves 0.44 m during the processing time of 64 ms in executing STFTs with 32 samples for a 500-fps video; it is 37 times the length of its body.

Considering these issues in pixel-level digital processing, our HFR-video-based activity sensing in this study, is designed with two-step pixel-level STFTs for the precise estimation of the time-variant flapping frequency responses of individual flying honeybees as well as their locations and velocities, when numerous honeybees are observed in the field of view of an HFR camera by executing the following processes as illustrated in Figure 5.1.

5.3 Proposed algorithm

On the basis of the concept in Chapter 5.2, we implement a pixel-level signal processing algorithm in this Chapter, with a two-step pixel-level STFT method for HFR-video-based honeybee activity sensing, in which the 1st pixel-level STFTs are used for vibration source localization and the 2nd pixel-level STFTs are used for precise TFR computation. The details are provided in the following subsections.

5.3.1 Pixel-level vibration source localization

The input images of $M \times N$ pixels are acquired at time $k\tau$ (frame number k and frame cycle time τ) as $I(x, y, k\tau)$, wherein the frame rate is $f_0 = 1/\tau$. The input images are converted to the TFR image by executing the STFT with $K_1 (= 2K'_1)$ samples:

$$\begin{aligned} \boldsymbol{F}(x, y, t) &= (F_0(x, y, t), \cdots, F_{K'_1-1}(x, y, t)), \\ &= \mathrm{STFT}(I(x,y, t - K'_1\tau), \cdots, I(x,y, t + (K'_1 - 1)\tau), \end{aligned} \tag{5.1}$$

where K_1 is the number of samples in the STFT for the input images, and $F_K(x, y, t)$ indicates the frequency component image at a frequency of $f_K = f_0 K/K_1$ $(K = 0, \cdot\cdot\cdot, K'_1{-}1)$.

The absolute value image of the cut-off TFR image $\boldsymbol{F}'_{k_1}(x, y, t)$, in which frequency components less than f_{k_1} are cut off from $\boldsymbol{F}(x, y, t)$, is normalized by that of the DC-cut-off TFR image $\boldsymbol{F}'_{k_0}(x, y, t)$, whose cut-off frequency is f_{k_0} as the following high-pass-filter image:

$$H(x, y, t) = \frac{|\boldsymbol{F}'_{k_1}(x, y, t)|}{|\boldsymbol{F}'_{k_0}(x, y, t)|}, \tag{5.2}$$

where k_i $(i = 0, 1)$ is an integer to set the cut-off frequency $f_{k_i} = f_0 k_i / K_1$ of the cut-off TFR image as follows:

$$\boldsymbol{F}'_{k_i}(x, y, t) = (0, \cdots, 0, F_{k_i}(x, y, t), \cdots, F_{K'_1-1}(x, y, t)) \tag{5.3}$$

The vibration candidate region is detected pixel-wise by checking the DC-cut-off TFR image $\boldsymbol{F}'_{k_0}(x, y, t)$ and the high-pass-filter image $H(x, y, t)$ as follows:

$$V(x, y, t) = \begin{cases} 1 & (H(x, y, t) \geq \theta_H, |\boldsymbol{F}'_{k_0}(x, y, t)| \geq \theta_F) \\ 0 & \text{(otherwise)} \end{cases}, \tag{5.4}$$

where θ_F is a threshold to determine whether the amplitude of the DC-cut-off TFR image is detectable. θ_H is a threshold to determine whether the high-pass-filter image, which indicates the ratio of the frequency components greater than or equal to the cut-off frequency f_{k_1}, is sufficiently large.

5.3.2 Individual honeybee labeling

The zeroth- and first-moment features are calculated for $M'N'$ subregions Γ_{ab} of $m \times n$ pixels in $V(x, y, t)$ as

$$M_{pq}(\Gamma_{ab}, t) = \sum_{x=am}^{a(m+1)-1} \sum_{y=bn}^{b(n+1)-1} x^p y^q \cdot V(x, y, t)\,, \tag{5.5}$$

where $M = mM'$ and $N = nN'$; p and q are non-negative integers to determine the order in moment feature calculation, satisfying $p + q \leq 1$. Γ_{ab} $(a=0,\cdots, M'-1, b=0,\cdots, N'-1)$ is expressed as

$$\Gamma_{ab} = \{(x, y)|\ (am+s, bn+t), 0 \leq s < m, 0 \leq t < n\}. \tag{5.6}$$

To localize multiple honeybee regions in $V(x, y, t)$, the connected components labeling process is accelerated to obtain the labeled honeybee regions with the cell-based labeling algorithm using the moment features for $M' \times N'$ subregions [**?**]. For the labeled honeybee regions ${}^{l}O(t)$ $(l = 0,\cdots, L-1)$ in $V(x, y, t)$, the label-domain moment features $M_{pq}({}^{l}O(t))$ $(p+q \leq 1)$ are sequentially accumulated, concurrently with the scanning of a flag map $P_{ab}(t)$ of $M' \times N'$ subregions:

$$M_{pq}({}^{l}O(t)) = \sum_{(x,y)\in{}^{l}O(t)} x^p y^q \cdot V(x, y, t), \tag{5.7}$$

where $P_{ab}(t)$ $(a=0,\cdots, M'-1, b=0,\cdots, N'-1)$ is defined for each subregion Γ_{ab} by checking $M_{00}(\Gamma_{ab}, t)$ as follows:

$$P_{ab}(t) = \begin{cases} 1 & (M_{00}(\Gamma_{ab}, t) \geq \theta_P) \\ 0 & \text{(otherwise)} \end{cases}, \tag{5.8}$$

and θ_P is a threshold to determine whether the subregion Γ_{ab} is used for cell-based labeling.

The center positions of the labeled honeybee regions ${}^lO(t)$ are calculated as follows:

$$ {}^l\boldsymbol{x}(t) = ({}^lx(t), {}^ly(t)) = \left(\frac{M_{10}({}^lO(t))}{M_{00}({}^lO(t))}, \frac{M_{01}({}^lO(t))}{M_{00}({}^lO(t))} \right). \tag{5.9} $$

The detailed processes used to calculate the label-domain moment features in an image are given in [19, 20].

5.3.3 Frequency response computation for tracked honeybees

The labeled honeybee region ${}^lO(t)$ is defined as a sub-region of $R\times R$ pixels so that the honeybee locates in its center, and its ROI image ${}^lI(x', y', t)$ of $R\times R$ pixels is selected from the input image as

$$ {}^lI(x', y', t) = I(x' + {}^lx(t) - R/2, y' + {}^ly(t) - R/2, t), \tag{5.10} $$

where (x', y') indicates the local coordinate system in the ROI image of $R\times R$ pixels.

For the ROI images, the image intensities at pixel (x', y') are converted to the TFR ROI images by executing the STFT with $K_2(= 2K'_2)$ samples as follows:

$$ \begin{aligned} {}^l\boldsymbol{F}(x', y', t) &= ({}^lF_0(x', y', t), \cdots, {}^lF_{K'_2-1}(x', y', t)), \\ &= \mathrm{STFT}({}^lI'(x', y', t - K'_2\tau), \cdots, {}^lI'(x', y', t + (K'_2 - 1)\tau), \end{aligned} \tag{5.11} $$

where K_2 is the number of samples in the STFT for the ROI images, and where ${}^lF_K(x', y', t)$ $(l = 0, \cdots, L-1)$ indicates the frequency component of the TFR ROI image at a frequency of $f'_K = f_0 K / K_2$ $(K = 0, \cdots, K'_2 - 1)$.

The vibration pixels in the ROI images, ${}^lV(x', y', t)$ $(l = 0, \cdots, L-1)$, are extracted when the peak frequencies are greater than or equal to the cut-off frequency f_{k_2}, and their TFR images at the peak frequencies are greater than or equal to a threshold θ'_F to determine whether the frequency spectrum peak is distinct:

$$ {}^lV(x',y',t) = \begin{cases} 1 & ({}^lf_{pk}(x',y',t) \geq f_{k_2}, {}^lF_{{}^lk_{pk}}(x',y',t) \geq \theta'_F) \\ 0 & \text{(otherwise)} \end{cases}, \tag{5.12} $$

where k_2 is an integer to set the cut-off frequency $f_{k_2} = f_0 k_2 / K_2$. The peak frequency in the ROI image ${}^lf_{pk}$ $(l = 0, \cdots, L-1)$ for the l-th labeled honeybee region is determined by searching the maximum value in the frequency range above the cut-off frequency f_{k_2} as follows:

$$ {}^lk_{pk}(x',y',t) = \mathop{\arg\max}_{k_2 \leq K \leq K'_2 - 1} {}^lF_K(x',y',t), \tag{5.13} $$

where ${}^lk_{pk}(x',y',t)$ $(l = 0, \cdots, L-1)$ is an integer to set the peak frequency ${}^lf_{pk} = f_0 \cdot {}^lk_{pk} / K_2$ at the pixel (x', y') in the ROI image.

To quantify whether there is a certain vibration component of honeybee flapping, the averaged frequency response ${}^l\boldsymbol{F}(x',y',t)$ $(l = 0, \cdots, L-1)$ is computed by averaging the TFRs at the extracted vibration pixels in the ROI images as follows:

$$ {}^l\bar{\boldsymbol{F}}(t) = \begin{cases} \dfrac{1}{{}^lS(t)} \displaystyle\sum_{(x',y') \in {}^lO(t)} {}^lV(x',y',t) {}^l\boldsymbol{F}(x',y',t) & ({}^lS(t) \geq \theta_S) \\ \boldsymbol{0} & \text{(otherwise)} \end{cases}, \tag{5.14} $$

where ${}^lS(t)$ $(l = 0, \cdots, L$–$1)$ is the number of the vibration pixels in the l-th labeled honeybee region ${}^lO(t)$,

$$ {}^lS(t) = \sum_{(x',y') \in {}^lO(t)} {}^lV(x',y',t), \tag{5.15} $$

and θ_S is a threshold to determine whether the wing-flapping of honeybees is observable.

Independently of the brightness amplitude and the low frequency component in the background scene, the normalized TFR ${}^l\boldsymbol{A}(t)$ $(l=0,\cdots,L-1)$ is computed in the range of their flapping frequencies by normalizing the cut-off TFR image,

$$ {}^l\bar{\boldsymbol{F}}'_{k_2}(t) = (0,\cdots,0,{}^l\bar{F}'_{k_2}(t),\cdots,{}^l\bar{F}'_{K'_2-1}), \tag{5.16} $$

whose cut-off frequency is f_{k_2}, as follows:

$$ {}^l\boldsymbol{A}(t) = {}^l\bar{\boldsymbol{F}}'_{k_2}(t) / \left| {}^l\bar{\boldsymbol{F}}'_{k_2}(t) \right| . \tag{5.17} $$

Assuming that the flapping frequency of a honeybee is larger than a certain frequency f_H, the flying activity of the l-th labeled honeybee is computed as its flapping frequency ${}^l f_A(t)$ $(l=0,\cdots,L\text{–}1)$ by detecting the peak frequency of ${}^l\boldsymbol{A}(t) = ({}^lA_0(t),\cdots,{}^lA_{K'_2-1}(t))$ in the following frequency range,

$$ {}^l f_A(t) = \frac{f_0}{K_2} \mathop{\arg\max}_{k_H \le K \le K'_2-1} {}^lA_K(t), \tag{5.18} $$

where k_H is an integer to set the frequency $f_H = f_0 k_H / K_2$, which determines the frequency range to be searched.

5.4 Sensitivity evaluation

First, we checked how the spatial resolution of an HFR video affects the sensitivity in our honeybee activity sensing algorithm when the image intensity at every pixel is converted to a TFR over a period of time. We captured 8-bit color images of 1024×1024 pixels at 500 fps ($\tau = 2$ ms) with a 0.5-ms-camera exposure when 12–14-mm size honeybees flew around their beehive in a cluttered outdoor location. An HFR camera was installed at a distance of 1.5 m from the beehive. The beehive was observed in the depth of field (DOF) of the HFR camera using 1024×1024 pixels, and the DOF far limit was set to infinity with a small lens aperture. A 1024×1024 image corresponded to a 0.6×0.6-m-area around the beehive; one pixel corresponded to 0.6 mm. In the evaluation, we used HFR videos for 0.256 s of 64×64, and 256×256 pixels ($M = N = 64$ and 256), as well as the original 1024×1024 HFR video ($M = N = 1024$), by interleaving the original HFR video at every 16 and 4 pixel, respectively; the pixel interval in each resolution corresponded to 9.6 and 2.4 mm, respectively. The parameters for step (1) are set as $K_1 = 32$, $f_{k_0} = 31.25$ Hz, $f_{k_1} = 62.5$ Hz, $\theta_F = 400$, and $\theta_H = 0.5$, those for step (2) are set as $m = n = 4$, 16, and 64, and $\theta_P = 1$, 5, and 20 when $M = N = 64$, 256, and 1024, respectively, and those for step (3) are set as $R = 4$, 16, and 64, when $M = N = 64$, 256, and 1024, respectively, $K_2 = 128$, $f_{k_2} = 62.5$ Hz, $\theta'_F = 400$, $\theta_S = 1$, and $f_H = 150$ Hz. These parameters are summarized in Table 5.1.

When the frequency range to be inspected is priorly known for wing-flapping of honeybees, K_1, f_{k_0}, f_{k_1}, K_2, f_{k_2}, and f_H, which are used to determine the frequency range and the number of samples in computing TFRs with pixel-level STFTs, can be set as universal values. They are independent of image resolution, image contrast and brightness, and apparent sizes of honeybees in images; therefore, there is no need to determine them each time the camera is deployed. The parameters m, n, θ_P, R, and θ_S depend on the apparent sizes of honeybees in images, corresponding to camera parameters such as image resolution, magnification ratio, and camera-to-object distance. They are independent from time-varying image brightness and contrast in ambient sunlight; therefore, there is no need to adjust these parameters once the camera is deployed at a fixed location. The parameters θ_F, θ_H, and θ'_F that mainly check the intensities of the pixel-level TFRs, depend on image contrast and brightness, and background movement. Under time-varying sunlight condition and background scene, they should be adaptively determined such that honeybee regions are properly extracted as vibration pixels. However, in this study, we manually determined these parameters for short-span HFR videos when there was no

Table 5.1: Parameters used for the experiments in Subsection 5.4.

step	parameter	64×64 (M=N=64)	256×256 (M=N=256)	1024×1024 (M=N=1024)
(1)	K_1	32		
	f_{k_0}	31.25 Hz		
	f_{k_1}	62.5 Hz		
	θ_F	400		
	θ_H	0.5		
(2)	$m(=n)$	4	16	64
	θ_P	1	5	20
(3)	R	4	16	64
	K_2	128		
	f_{k_2}	62.5 Hz		
	θ'_F	400		
	θ_S	1		
	f_H	150 Hz		

large time-variation in image brightness.

Figure 5.3 shows (a) the input images $I(x, y, t)$, (b) the candidate vibration pixels $V(x, y, t)$, and (c) the locations of selected ROI images ${}^{l}I(x', y', t)$ for honeybees. Figure 5.4 shows the magnified ROI images ${}^{l}I(x', y', t)$ for "bee 1" and "bee 2", which are labeled in Figure 5.3. Figure 5.5 shows the averaged TFRs ${}^{l}\boldsymbol{A}(t)$ in the range from 62.5 to 250 Hz with a frequency resolution of 3.91 Hz, which were computed using the 2nd pixel-level STFTs for these ROI images to determine the flying activities of the labeled honeybees. "bee 1" was flying at a certain speed in the direction from right to left, and "bee 2" was hovering in the input images. Figure 5.6 plots (a) the temporal changes in the image intensities at $A(960, 294)$, $B(319, 609)$, and $C(270, 50)$ in the original input images of 1024×1024 pixels and (b) those at $A'(32, 32)$ and $B'(32, 32)$ in the tracked ROI images of 64×64 pixels for 0.2 s. Here, point A is around "bee 1", point B is around "bee 2", and point C is on a wooden box in the background as illustrated in Figure 5.3(a). Points A' and B' correspond to the centers of the tracked ROI images for "bee 1" and "bee 2", respectively, as illustrated in the tracked 64×64 ROI images in Figure 5.4. As shown in Figure 5.6, no rapid changes were observed in the image intensity at point C in the background scene. Only two or three pulses in a short-term duration were observed in the image intensities at points A and B around the honeybees; they were flying at a certain speed with their wings flapping at hundreds of Herz. In Figure 5.6(b), periodic waves with a duration of 0.2 s on the envelope waves, which are drifting with the background and illumination conditions around the honeybees, are observed at points A' and B' in the tracked ROI images; no data was plotted when the tracking was failed. This is because honeybee tracking enables the long-time observation of periodic changes in the image intensities at the pixels around flying honeybees.

In all the cases of 64×64, 256×256, and 1024×1024 input images, candidate vibration pixels were extracted around the flying honeybees with pixel-level digital signal processing in our algorithm. The honeybees were tracked correctly in the selected ROI images even when they can be hardly observed in a cluttered scene because their apparent sizes were 1 or 2 pixels in the 64×64 input images. The vibration region of “bee 1” had a comet-like shape, whose size is larger than the actual size of the honeybee because the honeybee was flying at a high speed during 64 ms in the STFT computation with $K_1 = 32$. The honeybee’s velocity was used to determine the direction and length of the comet shape. The vibration region of “bee 2”, hovering at the same place, had a small and non-directional shape, compared with that of “bee 1”.

In the averaged TFRs for 256×256 and 1024×1024 input images in Figure 5.5, significant frequency components were observed around 175 and 235 Hz, respectively. They corresponded to the different wing-flapping frequencies of “bee 1” and “bee 2”, respectively. Here, the peak frequency detection was conducted as a maximum search process, and it was not very sensitive even in the case of a low signal-to-noise ratio. This was because the small periodic waves corresponding to the wing flapping of the honeybees had a distinct frequency component at their wing-flapping frequencies, whereas the TFRs of the drifting envelope waves were almost flat in the frequency range for peak frequency detection because they did not have any specific frequency component. Their peak frequencies were not apparently observed in both the averaged TFRs for “bee 1” and “bee 2” for 64×64 input images because the locations of the selected ROIs fluctuated pixelwise at every frame. These fluctuations are not negligible in pixel-level digital signal processing when one pixel is similar or larger than the size of honeybee in the 64×64 input images. The spatial resolution of a honeybee in an image was inversely proportional to its distance from the camera. Honeybees at a distance of 24 m from the camera could be detected in a 1024×1024 image using pixel-level STFTs, which corresponded to a 9.6×9.6 m area with a 9.7 mm pixel pitch. This was because honeybees at a distance of 1.5 m from the camera were detected in 64×64 images, which corresponded to a 0.6×0.6 m area with the same pixel pitch.

These evaluation results indicate that honeybee tracking can be robustly executed by extracting around flying honeybees in an HFR video using the 1st pixel-level STFTs for vibration source localization, independent of the image resolution. In the meantime, input images with a certain spatial resolution, in which honeybees are apparently observed, are required for the 2nd pixel-level STFTs to stably obtain the frequency responses of flying honeybees to determine their flying activities.

5.5 Honeybee tracking experiment

To verify the trajectories and flying activities of honeybees in outdoor scene, 1024×1024 ($M = N = 1024$) input images, captured at 500 fps with a 0.5-ms camera exposure for 14 seconds, were analyzed with our activity sensing algorithm. We captured the 500-fps video of 1024×1024 pixels for the same scene in a 0.6×0.6-m-area around their beehive in a cluttered outdoor location as that had been analyzed in Subsection 5.4; one pixel corresponded to 0.6 mm. The parameters in our algorithm were set to the same ones as those set when $M = N = 1024$ in Subsection 5.4. In the experiment, most of the flying honeybees were wing-flapping at frequencies below the Nyquist frequency at 250 Hz.

Figure 5.7 shows (a) the 1024×1024 input image $I(x, y, t)$ at $t = 0$ s, (b) the xy-trajectories of the labeled honeybees ${}^{l}\boldsymbol{x}(t)$ for t = 0–14 s, and (c) the flapping frequencies of the labeled honeybees ${}^{l}f_A(t)$. In (b), the labels of “bee 1”–“bee 10” indicate the indexes in the labeling with moment feature accumulation. These indexes were updated as those of their nearest honeybees at the previous frame; abandoned indexes that were not assigned at the current frame, were reused when a newly labeled honeybee did not correspond with any labeled honeybees at previous frame. Each xy-trajectory involves many temporally-disconnected trajectories of different honeybees, because the duration times of their trajectories were mostly less than one second and several trajectories were crossed over each other. The peak frequencies of the labeled honeybees in (c) were color-mapped as their flapping frequencies along their trajectories.

It can be observed that there are many complex-shape trajectories of honeybees around their beehive, whereas there are gentle curved trajectories of honeybees in the horizontal direction at distant places from their beehive. Here, these trajectories did not involve the movements of the flying honeybees in the depth direction, because our method was limited to 2-D measurements in the horizontal and vertical directions. The complex-shape trajectories corresponded to those of honeybees hovering around their beehive, and the flapping frequencies on the trajectories were mostly over 220 Hz; most of them entered their beehive with hovering for a certain time when they flew back to their beehive. In this experiment, most of the gentle curve trajectories at distant places from their beehive corresponded to the honeybees which flew back to their beehive. The flapping frequencies on the horizontal trajectories at distant places were mostly less than 200 Hz, because the flapping frequencies of honeybees decreased for them to land at their beehives. The flapping frequencies were uncertain at several segments on the honeybee trajectories due to the unstable TFRs in brightness with the crosstalk background patterns and the transparency of honeybee wings. We note that the wing-flapping of a honeybee at a frequency above the Nyquist frequency is undersampled as the low-frequency alias. If a honeybee is flying with wing-flapping in the range from 250 to 300 Hz in a 500-fps video, it is detected as an undersampled component in the range from 200 to 250 Hz; it is distinctively separable from the background DC component in vibration source localization.

Figure 5.8 shows (a) the 64×64-input images ${}^{l}(x', y', t)$ in the tracked ROI around "bee 2", (b) the vibration pixels ${}^{l}V(x', y', t)$ and (c) the peak frequencies ${}^{l}f_{pk}(x', y', t)$ in the tracked ROI, (d) the time-transient averaged TFR ${}^{l}\boldsymbol{A}(t)$ in the range of 62.5 to 250 Hz, (e) the apparent flying speed $|{}^{l}\boldsymbol{v}(t)|{=}|d\,{}^{l}\boldsymbol{x}(t)/dt|$ and flapping frequency ${}^{l}f_{A}(t)$ of the labeled honeybee, and (f) the flapping frequency color-mapped on its xy-trajectory when "bee 2" flew back to its beehive from right to left for t = 6.36–6.96 s; (a)–(c) were obtained at intervals of 0.20 s, and the moving averages of the speeds and flapping frequencies among continuous 10 frames were plotted in (e).

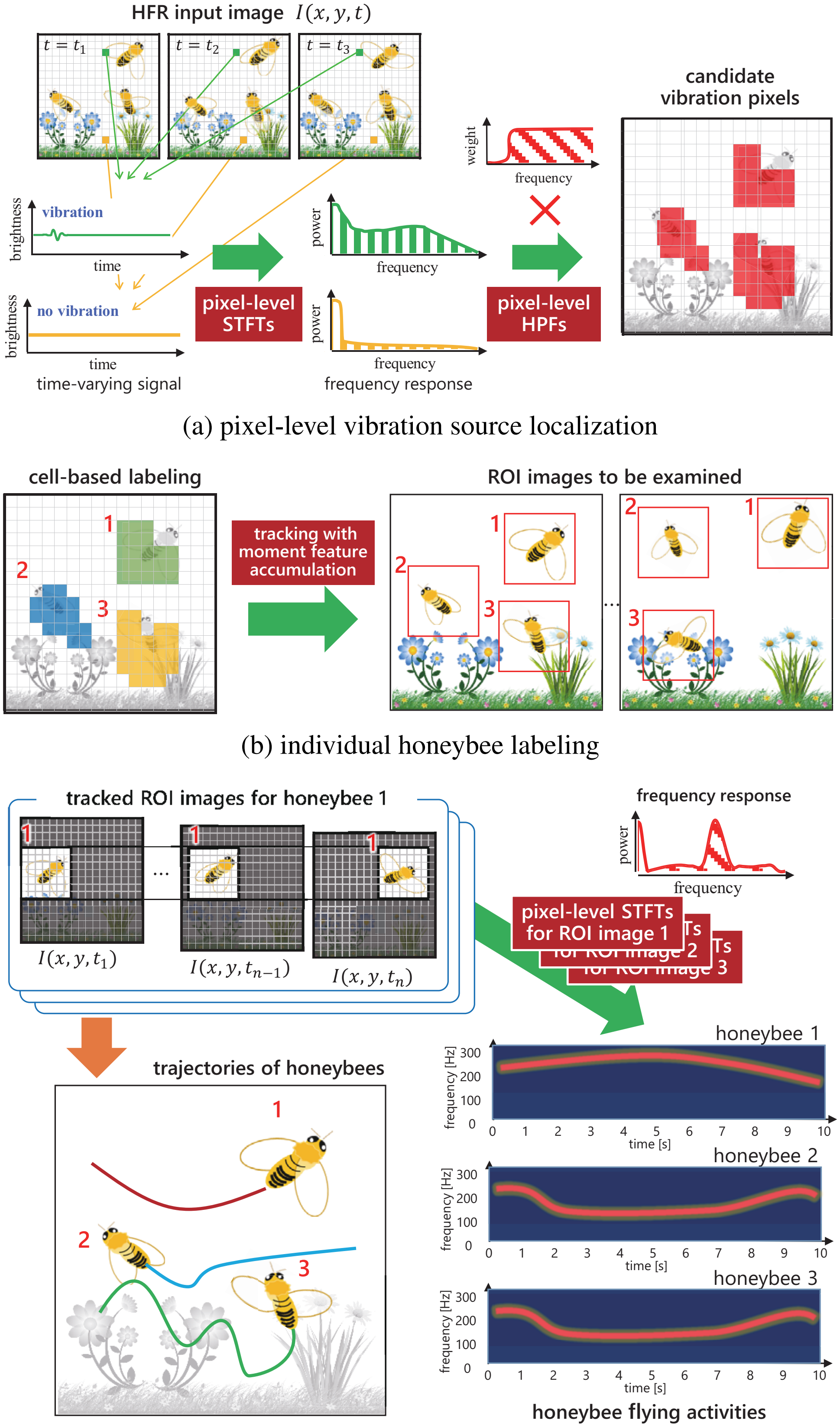
In Figure 5.8, the apparent flying speed of the honeybee decreased from 3.3×10^{3} pixel/s at t = 6.36 s to 2.7×10^{2} pixel/s at t = 6.96 s in the input images. The apparent speeds correspond to 2.0 and 0.14 m/s, respectively, around its beehive when the velocity components in the depth direction are ignored; the actual speeds of the honeybees flying in 3-D space are larger than these apparent speeds. It can be observed that a flying honeybee was always in the tracked 64×64 ROI images, and the vibration pixels were detected around its flapping wings by detecting the peak frequencies in the tracked ROI. Compared with the results in Figure 5.3(b), the vibration pixels have no comet-like shape because the latency effect in digital signal processing when observing fast-flying honeybees was remarkably reduced in the tracked ROI images by canceling the translational movements of honeybees in the original input images. Corresponding to the wing-flapping frequency of the honeybee, a significant frequency component around 175 Hz was observed in the averaged TFR when the honeybee was flying at 1 m/s or more. Its peak frequency was fluctuated for t = 6.65–6.85 s because the crosstalk in the averaged TFR was generated by apparently-moving background patterns in the tracked ROI images.

In a similar manner, Figure 5.9 and 5.10 show (a) the tracked ROI images, (b) the vibration pixels, (c) the peak frequencies, (d) the averaged TFR, (e) the honeybee speed and flapping frequency, and (f) the xy-trajectory with flapping frequency when "bee 1" was hovering around its beehive for t = 7.64–8.24 s, and when "bee 1" took off and flew away from its beehive for t = 8.82–9.12 s, respectively; the honeybee labeled as "bee 1" in Figure 5.9 was different from that labeled as "bee 1" in Figure 5.10. When the velocity components in the depth direction are ignored, the apparent flying speed of a honeybee when hovering in Figure 5.9 was less than 1.6×10^2 pixel/s, corresponding to 0.10 m/s in the actual scale. The apparent speed when flying away in Figure 5.10 increased from no translation at t = 8.82 s to 7.2×10^3 pixel/s at t = 9.12 s in input images, corresponding to 4.3 m/s in the actual scale. In both cases, flying honeybees were stably tracked in the ROI images, and the vibration pixels were detected around their flapping wings as well as those in Figure 5.8. The flapping frequency when a honeybee was hovering was around 235 Hz in Figure 5.9, and that when a honeybee flew away from its beehive in Figure 5.10 increased from 150 to 205 Hz. The peak frequency in Figure 5.10 was misdetected for t = 8.82–8.85 s and 8.92–8.98 s with background crosstalk in the tracked ROI images. It can be observed that the flapping frequency of a honeybee time-varied depending on its flight status; the largest flapping frequency was observed when the honeybee was hovering, followed by that when the honeybee was flying away from its beehive, and that when the honeybee was flying back to its beehive.

These experimental results show that our HFR-video-based algorithm has pixel-level acute sensitivity to localize and track small flying honeybees in a cluttered outdoor scene where it is challenging for human eyes to discern them. It has the potential to quantify individually the wing-flapping frequencies of flying honeybees at audio frequency level, because it inspects their time-varying TFRs in all the tracked ROIs as their flight activity indexes.

5.6 Concluding remarks

In this chapter, we proposed a novel honeybee activity sensing algorithm that can track and quantify the vibration properties in brightness at the pixel around wing-flapping honeybees by executing two-step pixel-level STFTs for an HFR video. We conducted a tracking experiment for honeybees flying around their beehive in a cluttered scene by analyzing 1024×1024 images captured at 500 fps, and demonstrated the discriminative ability of the HFR video to detect the flapping frequencies of honeybees at hundreds of Hertz in the frequency range below 250 Hz as their flight activities, as well as its pixel-level localization ability for individual honeybees. Considering the frequency spectrum peaks are broad and their time-varying peak frequencies when different types of flying insects are observed in an HFR video, we intend to improve our algorithm to distinguish honeybees by collecting and analyzing both their appearances and time-varying frequency spectrums using machine learning technology for pattern recognition. We also intend to use an HFR video at thousands of fps such that the Nyquist frequency is much larger than the wing-flapping frequencies of insects to be observed, and develop a real-time HFR camera dedicated to long-time honeybee activity sensing by using graphic processing units to accelerate the computational speed of pixel-level STFTs in our algorithm.



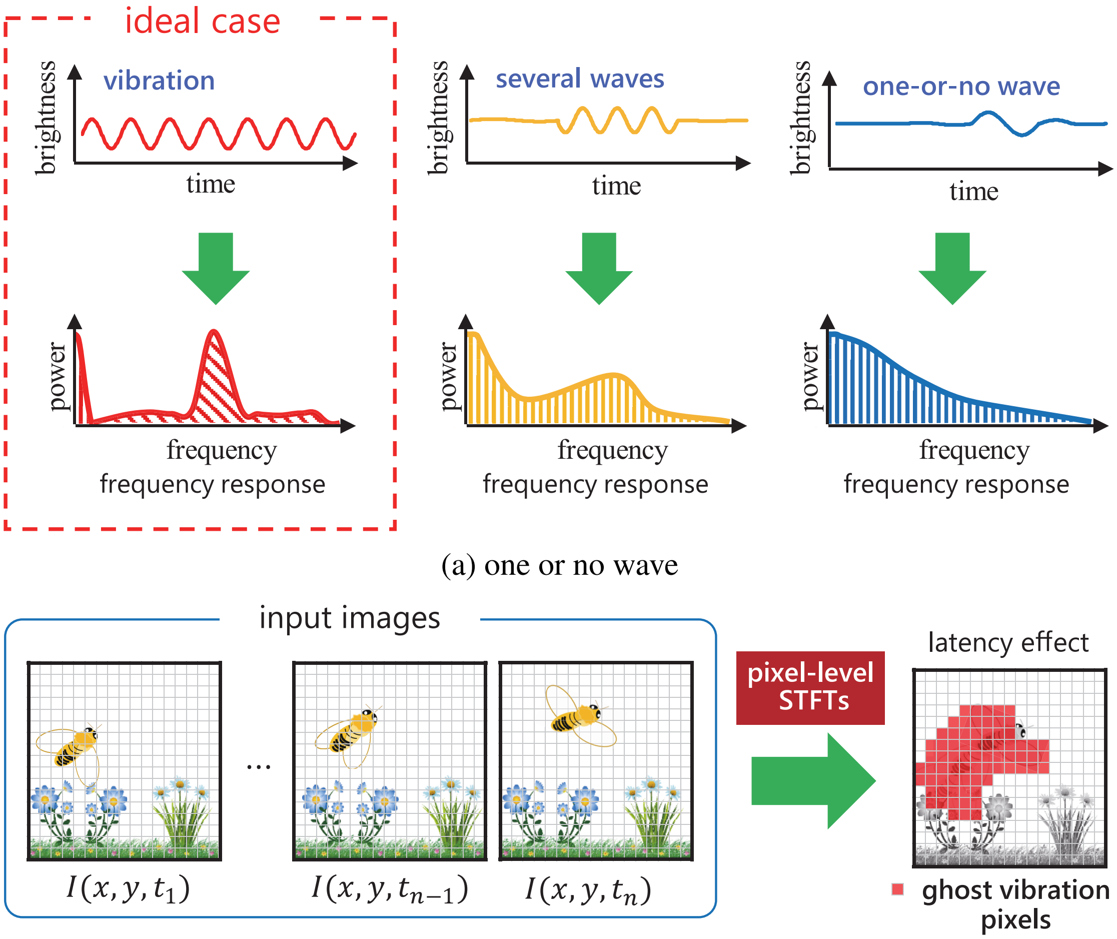
(a) pixel-level vibration source localization

(b) individual honeybee labeling

(c) frequency response computation for tracked honeybees

Figure 5.1: Concept of pixel-level flight activity sensing method

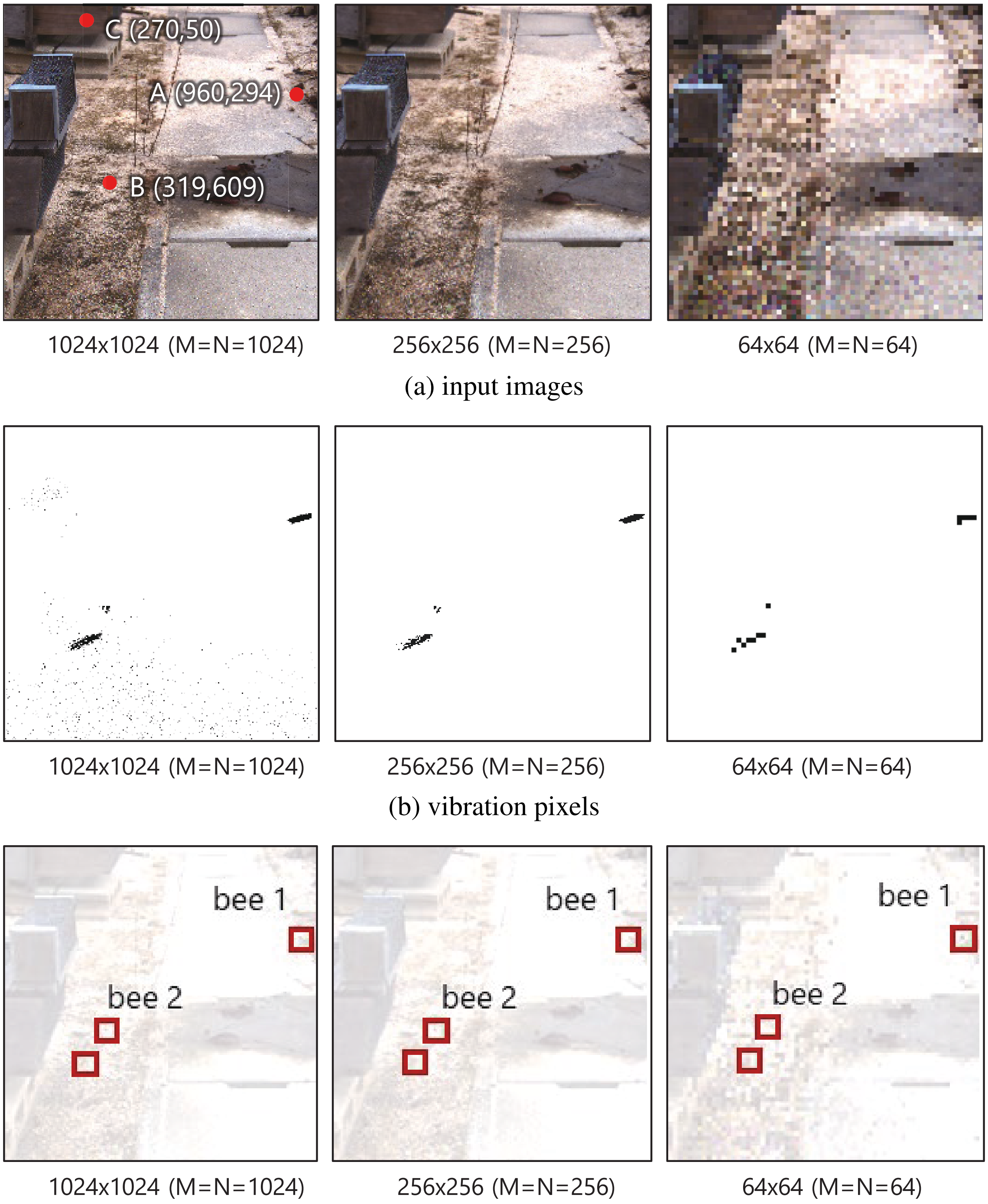
.



(a) one or no wave

(b) latency effect

Figure 5.2: Problems in pixel-level digital signal processing when observing fast-flying honeybees.

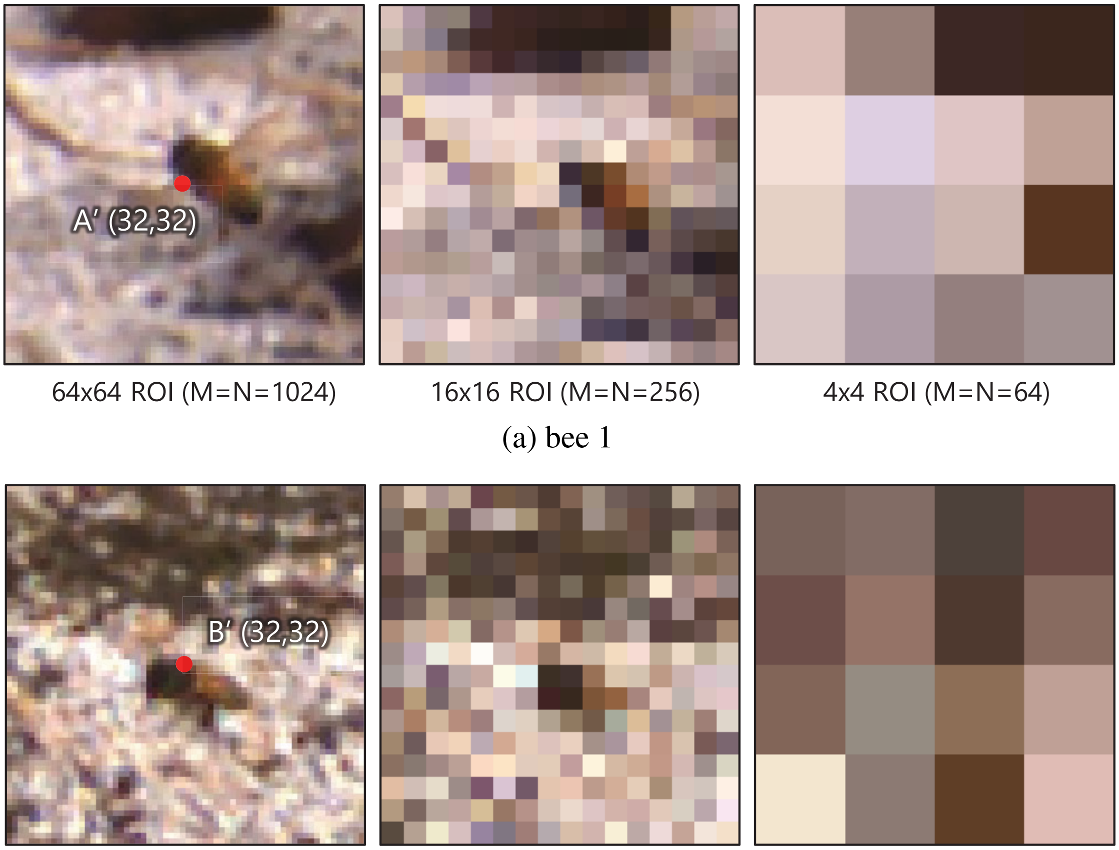


(a) input images

(b) vibration pixels

(c) locations of selected ROIs

Figure 5.3: Input images, candidate vibration pixels, and selected ROIs for honeybees in sensitivity evaluation.



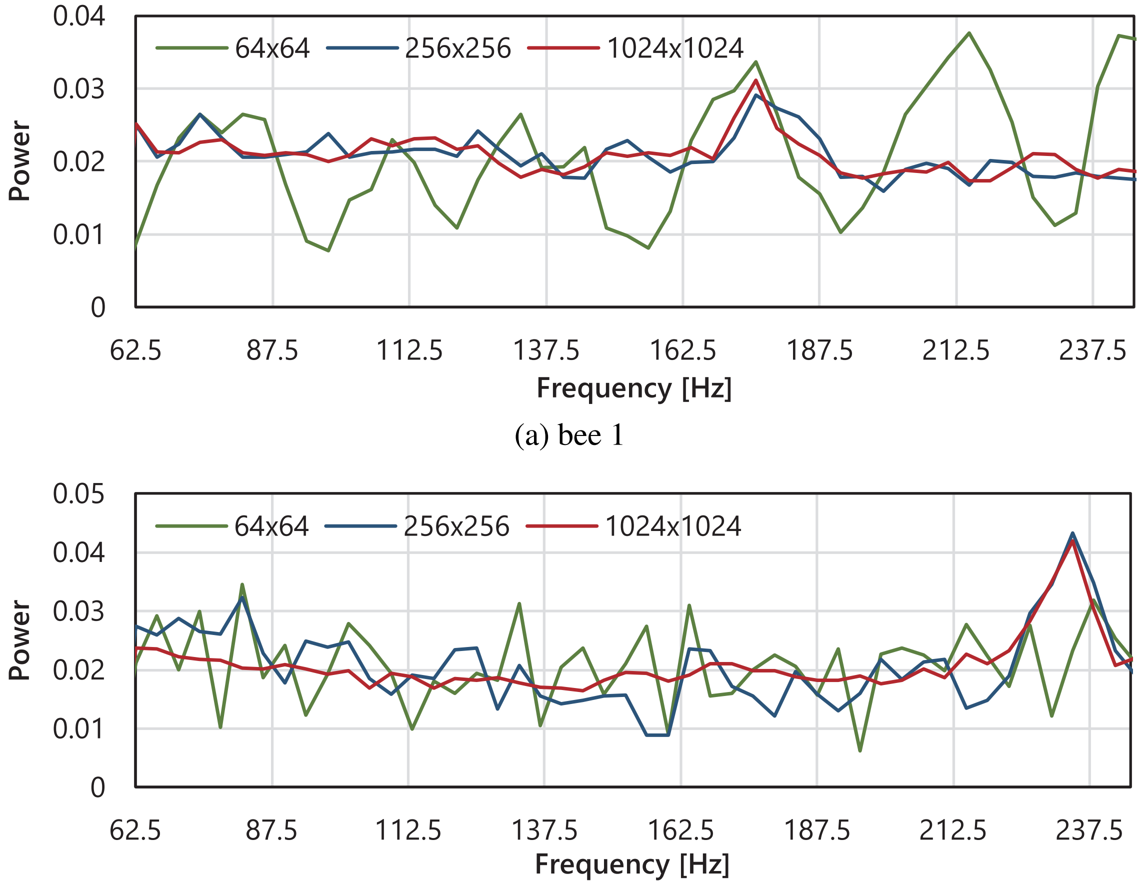
64x64 ROI (M=N=1024) 16x16 ROI (M=N=256) 4x4 ROI (M=N=64)

(a) bee 1

64x64 ROI (M=N=1024) 16x16 ROI (M=N=256) 4x4 ROI (M=N=64)

(b) bee 2

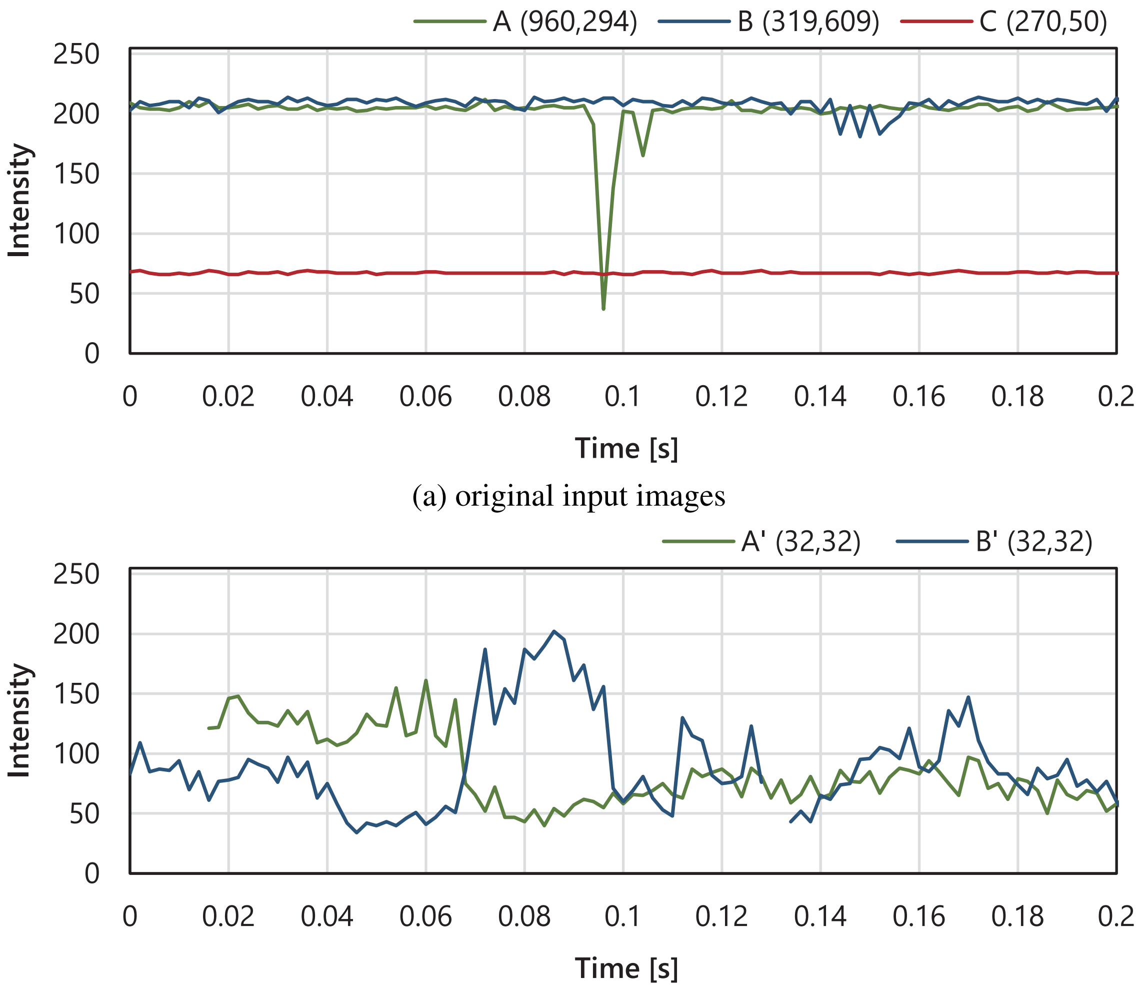
Figure 5.4: Magnified ROI images in sensitivity evaluation.



(a) bee 1

(b) bee 2

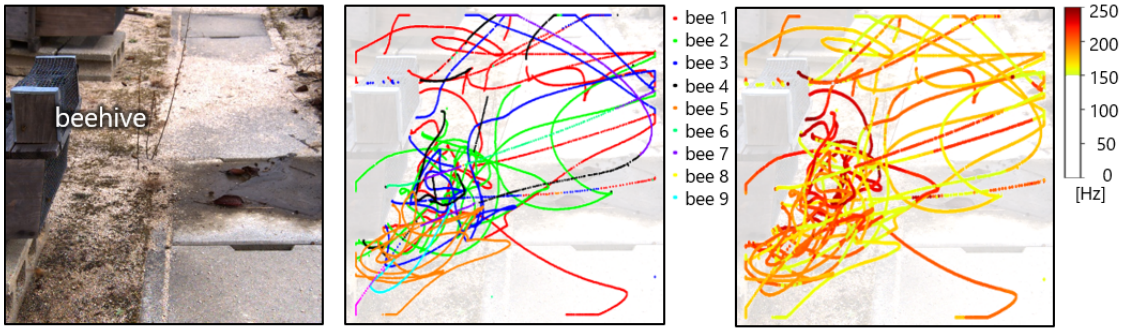
Figure 5.5: Averaged TFRs in the selected ROI images in sensitivity evaluation.



(a) original input images

(b) tracked ROI images

Figure 5.6: Temporal changes in image intensities in sensitivity evaluation.

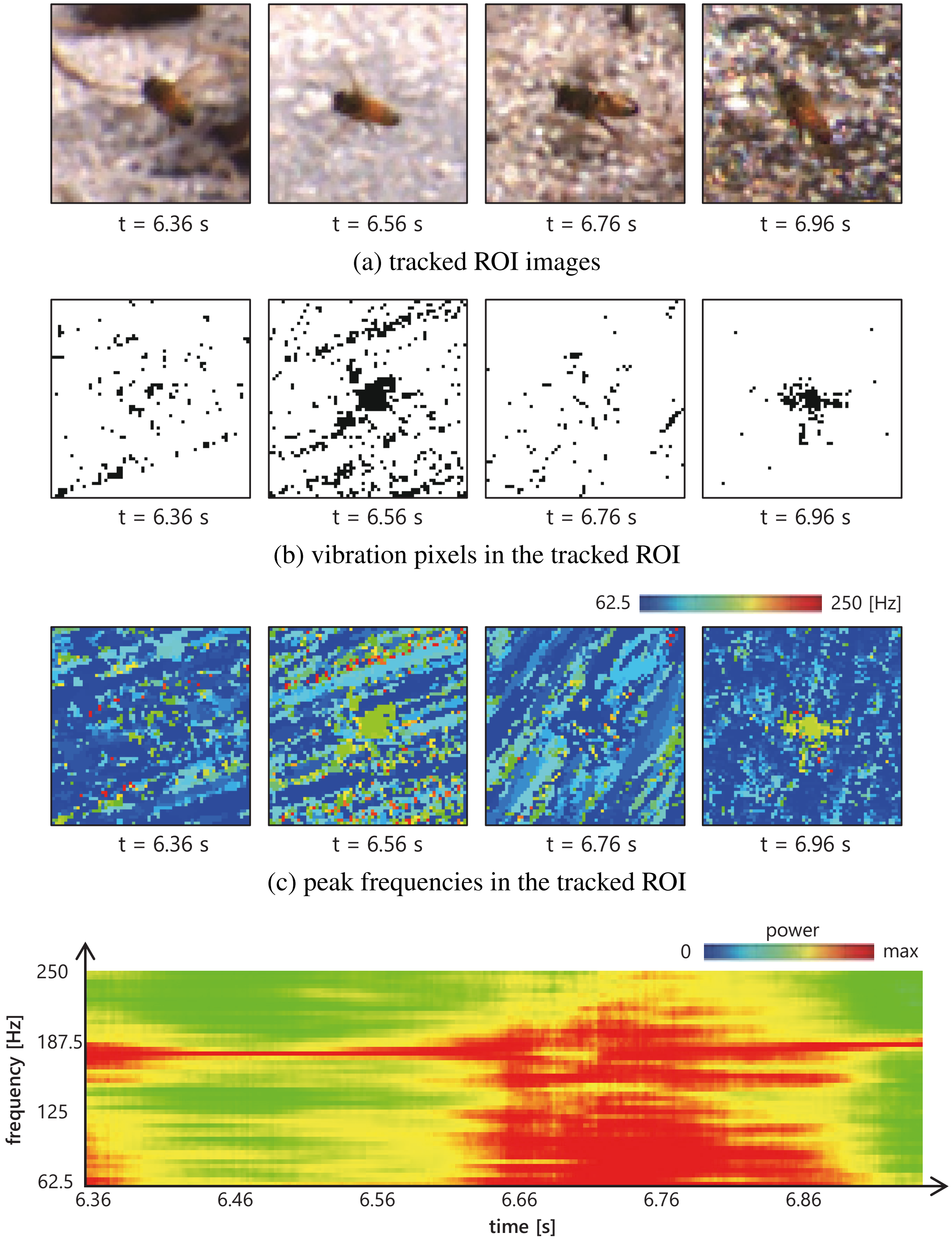


(a) input image

(b) *xy*-trajectories

(c) flapping frequencies

Figure 5.7: *xy*-trajectories and flapping frequencies of labeled honeybees in 14 seconds.

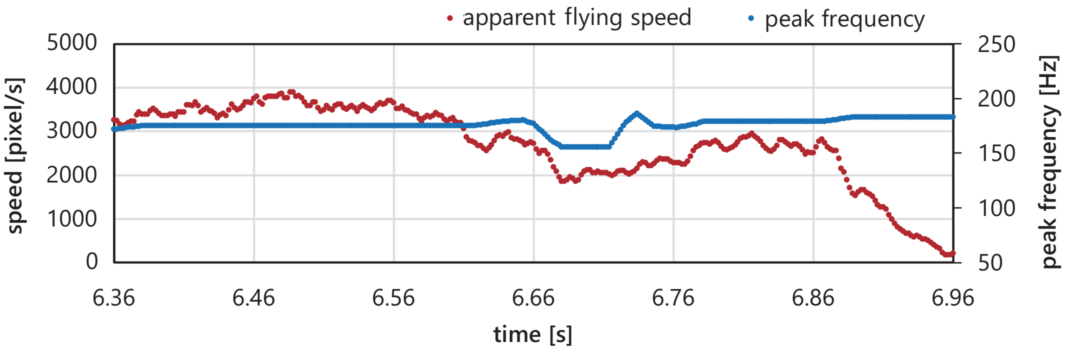


(a) tracked ROI images

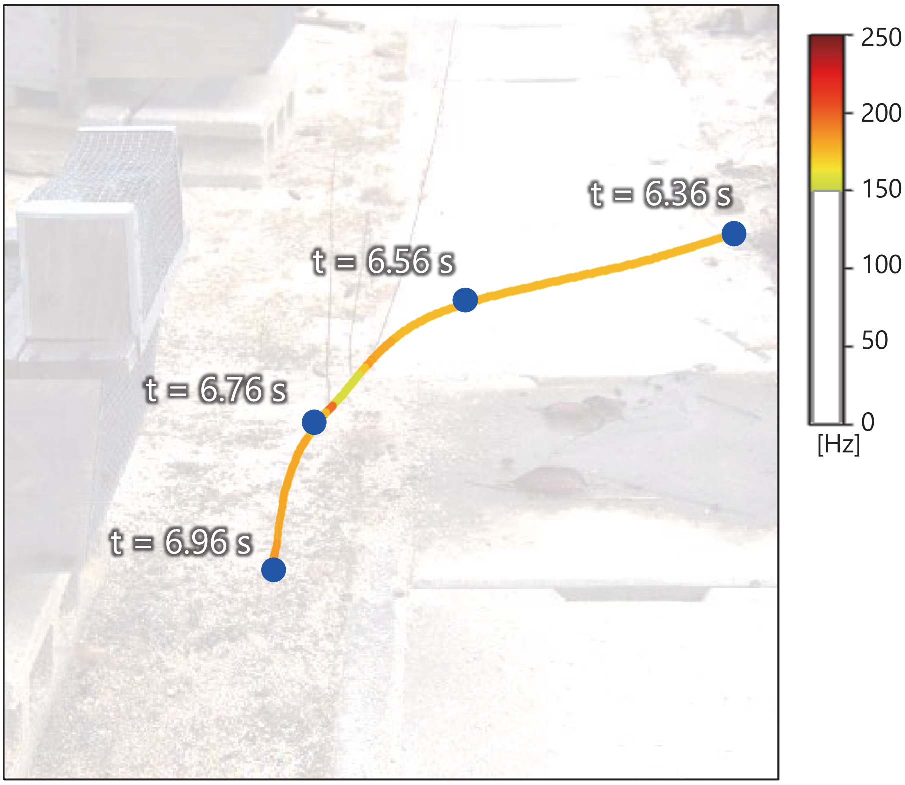
(b) vibration pixels in the tracked ROI

(c) peak frequencies in the tracked ROI

(d) time-transient averaged TFR

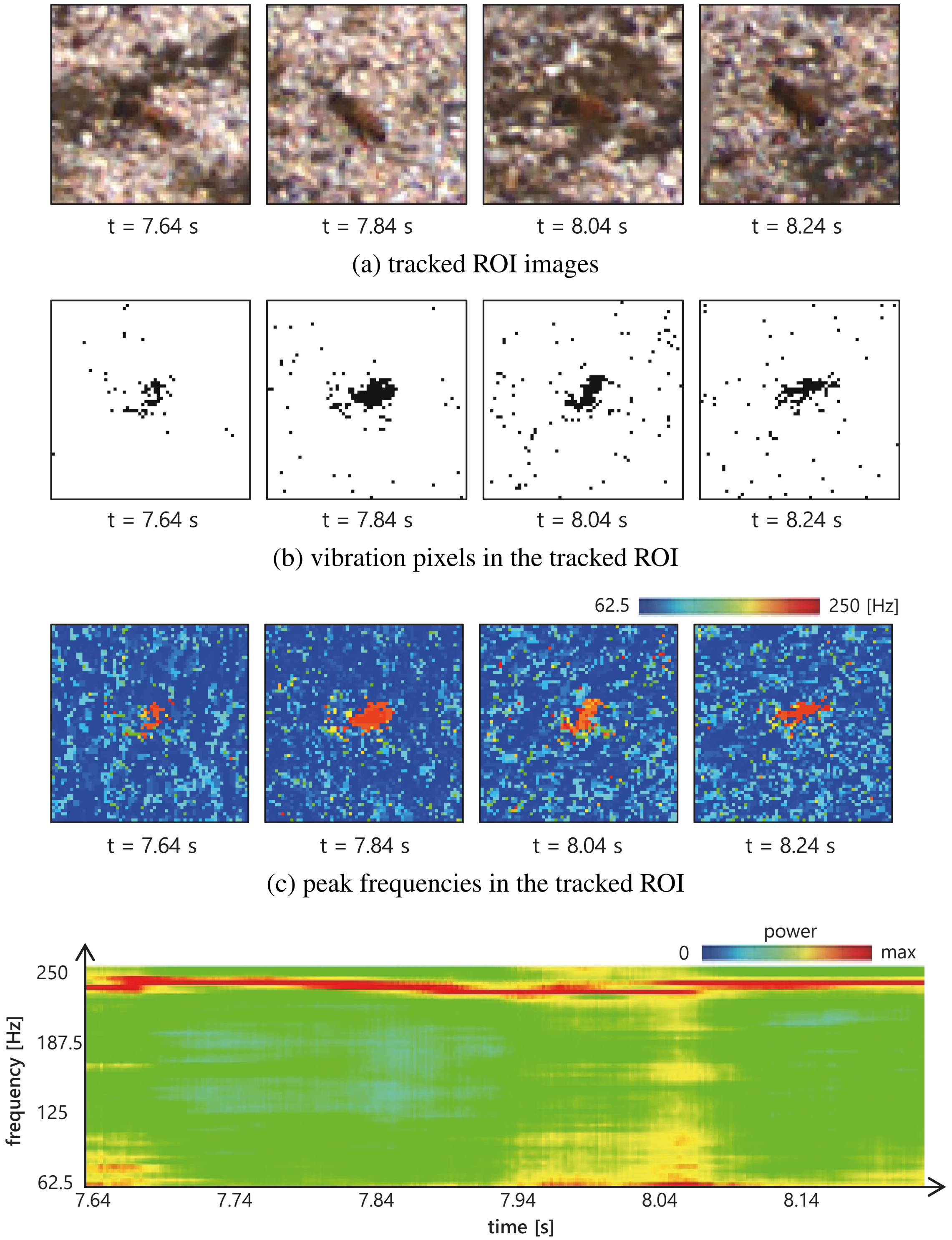


(e) apparent flying speed and flapping frequency



(f) flapping frequency mapped on 0.60-s *xy*-trajectory

Figure 5.8: Tracked ROI images and flying activities when "bee 2" flew back to its beehive for *t* = 6.36–6.96 s.

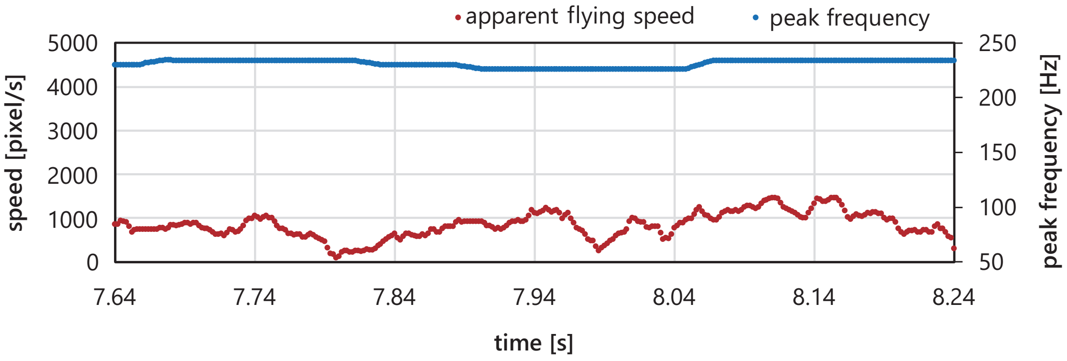


(a) tracked ROI images

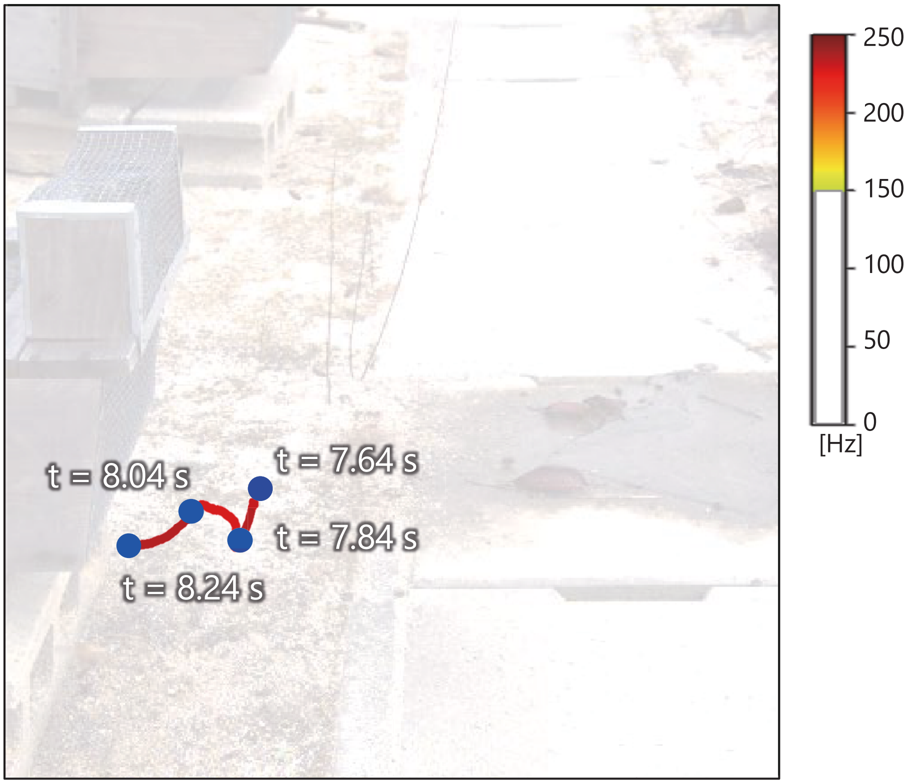
(b) vibration pixels in the tracked ROI

(c) peak frequencies in the tracked ROI

(d) time-transient averaged TFR

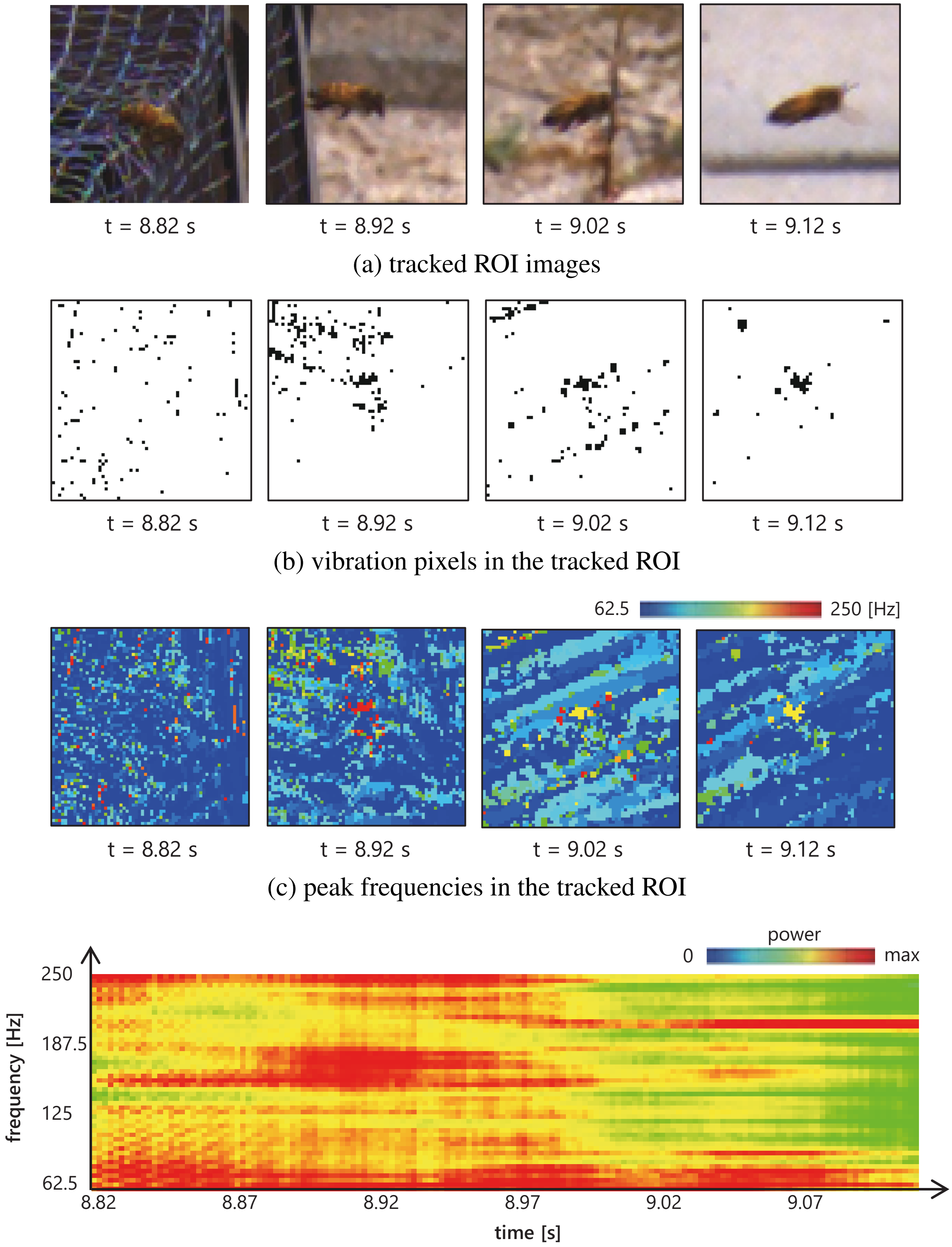


(e) apparent flying speed and flapping frequency



(f) flapping frequency mapped on 0.60-s xy-trajectory

Figure 5.9: Tracked ROI images and flying activities when "bee 1" was hovering around its beehive for t = 7.64–8.24 s.

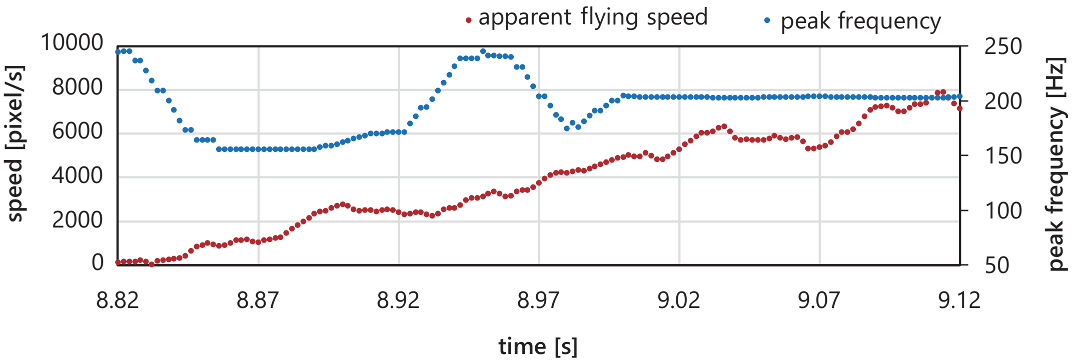


(a) tracked ROI images

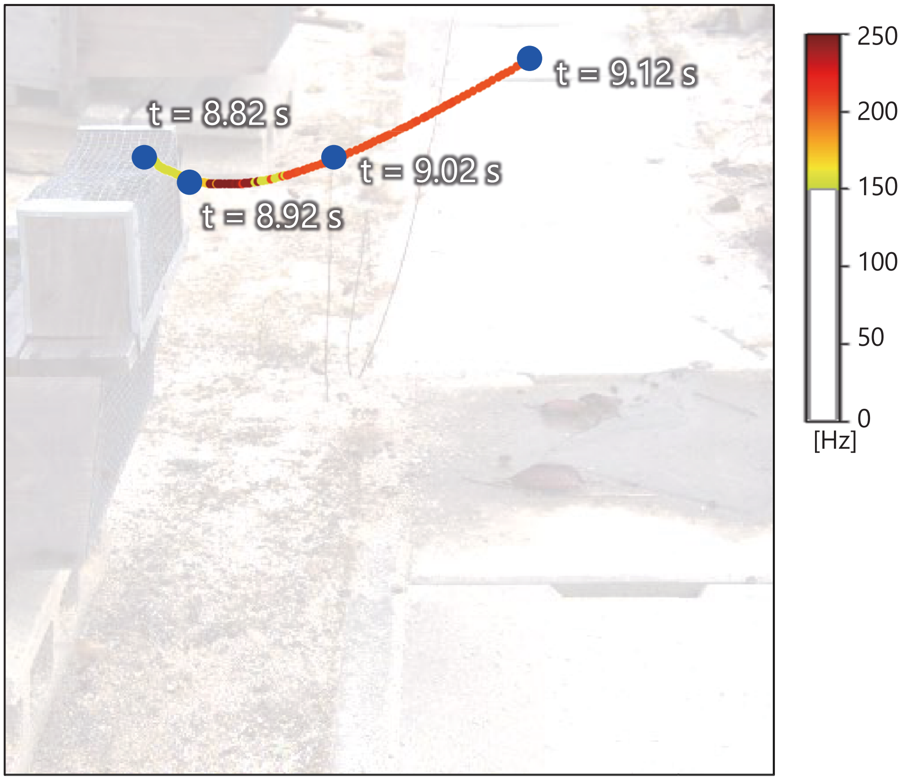
(b) vibration pixels in the tracked ROI

(c) peak frequencies in the tracked ROI

(d) time-transient averaged TFR



(e) apparent flying speed and flapping frequency



(f) flapping frequency mapped on 0.30-s *xy*-trajectory

Figure 5.10: Tracked ROI images and flying activities when “bee 1” flew away from its beehive for t = 8.82–9.12 s.

Chapter 6

Conclusions

In this study, to realize the visualization of vibration information from target objects whose dynamics property can be obtained by converting the time-varying signal measured by an audio or vibration sensor into a frequency in the frequency domain. Moreover, we developed a high-speed-vision-based pixel-level vibration spectrum imaging system in which STFTs and digital filters were implemented in parallel for the time-varying signals of all the pixels.

We proposed three concepts for the pixel-level vibration spectrum imaging. The total concept is the pixel-level vibration spectrum imaging method. When a vision system operates at a sufficiently high frame rate, temporal periodic changes can be observed in the brightness signals at the pixels around vibrating objects corresponding to their vibration's dynamic properties in the audio frequency range. The brightness signals' dynamic properties can be quantified in the form of a pixel-level distribution by converting the brightness signals for all the pixels of the HFR images into their temporal frequency responses using STFT. This method expands two types of applications for flying objects.

The first application proposes a fast algorithm for vision-based vibration source localization for flying multi-copter that can detect vibration sources at hundreds of Hertz by inspecting time-varying brightness signals at each pixel in HFR images. Our algorithm can significantly reduce the computational complexity of pixel-level digital filters for vibration source localization by virtually adjusting the sampling rate to twice the vibration frequency of a target object to be tracked using downsampled HFR images with frame interpolation while locking in vibration frequency for tracking objects.

The second application is pixel-level flight activity sensing method by utilizing pixel-level vibration source localization and vibration spectrum imaging for flying honeybee. This concept shows the flight activity sensing that can track and quantify the vibration properties in brightness at the pixel around wing-flapping honeybees by executing two-step pixel-level STFTs for an HFR video. In several experimental results, flying honeybees in 1024×1024 images captured at 500 fps demonstrate that, even when the image region of the honeybee is low resolution, robust tracking can be achieved with activity sensing in the frequency range up to 250 Hz, which is the Nyquist frequency when we analyze a 500 fps video. The trajectory of a detected and labeled honeybee is monitored with the peak frequency of these tracked STFTs as its flight activity.

This pixel-level STFT function for high-frame-rate images at 1000 fps or greater is accelerated through its parallel-implementation on a GPU-based high-speed vision system, and vibration spectrum images of 512×512 pixels can be obtained in real-time at more than 50 fps. Its effectiveness was shown through several verification experiments, including the vibration results for a fan, multi-copter, and guitar string, flying honeybee, which had rotating or vibrating frequencies in the range of hundreds of Hz. These results can show the possibility of applying for a real filed monitoring system.

The following issues remain to be solved in the future. The detection frequency resolution is an issue when they use high-speed vision at hundreds and thousands of fps. As we mentioned the Nyquist frequency and STFT samples, we determined frequency resolution between these relationships. After detecting target frequency in the proposed method, we implemented frame-locking into a target frequency to adjust the high accuracy of frequency resolution using the down-sampling method. Although we succeeded in developing a real-time vibration imaging system at 2000 fps, there is much higher frequency's phenomena that relate a natural frequency, a resonant frequency, and a chattering frequency and so on than 1000 Hz in several fields such as milling process and motor control process. Then, we need to plan to develop a real-time ultrafast vibration imaging system at 10000 fps or more. We also intend to develop a real-time HFR camera dedicated to long-time sensing by using GPU to accelerate the computational speed of pixel-level STFTs in our algorithm. Furthermore, we also plan to research mechanical vibration analysis, structural inspection, and biological dynamics inspection as applied research on real-time vibration spectrum imaging in future works.

Appendix

Onsite project in TAOYAKA program

Overview of our onsite team project

In this section, I summarize the activity of the Onsite-Team-Project in the TAOYAKA program, which was held from April 2018 to March 2019. I belong to the TAOYAKA program, which promoted how to learn reverse innovation during my master and Ph.D. course at Hiroshima University. In the TAOYAKA program, we made a group of Onsite-Team-Project consisting of three members, which take up the challenge resolving multicultural coexistence issues in a disadvantaged area, from a multilateral perspective of cultural creation, technical creation, and social implementation. Our Onsite-Team-Project focused on mobile application based rural development to promote sightseeing in Iinan town in Shimane prefecture. The sightseeing of hilly and mountainous areas had an essential role in Japan. However, it faces the common issues such as lack of labor and aging of labor and lack of tourists. As the Japanese government would like to implement revitalization efforts outside of Tokyo, this area is drastically declining in population, which has affected the economy and the social situation. Aging Population of Iinan Town, the population is significantly growing in elderly residents. Our team has identified that Iinan Town would greatly benefit from a social and economic revitalization project due to its population decline, low level of tourism, and current economic situation. We proposed the social and economic revitalization of Iinan Town community through the development and implementation of a mobile application using Augmented Reality (AR) with embedded video camera and geolocation. It allows players to visit various rural community tourist spots and interact with rural community members by playing game story (currently prototype stage).

This mobile application is explained as follows: When the player launches this application, a mystery map launches the phone's GPS, which will then provide directions to the mystery rural town tourist spots. Players can see these cats only by looking through this application interface. Different interactions allow you to befriend the virtual character cats. When the player visits the tourist spot and scans the QR code at the tourist location, a cat then appears through AR. The Players visit the tourist spots to learn more about how to befriend the cats and find them by talking with community members. The objective of this application is for players from all over Japan to visit the rural community to interact with locals while visiting tourist spots and spend money within the community. Moreover, we propose the sustainable application development through the mobile application to focus on tourism regarding young generation education and I-turn and U-turn to increase the population. Increasing tourism in rural communities not only boosts the social interaction among the elderly in the community but will possibly boost the economy of the community.

For my part as a member of the Onsite-Team-Project, I assisted with the development of a mobile application game using AR technology with integrated geo-location tagging software to promote tourism, economic growth, and stimulate social interaction for Iinan Town. Our team cooperated with the University of Texas at Dallas, Narrative Systems laboratory in the USA.

Through this activity, I suggested that we promote a programming school as this mobile application can be a sustainable development for the local community.

If local people continue to develop this application, they can get the benefits to pro-

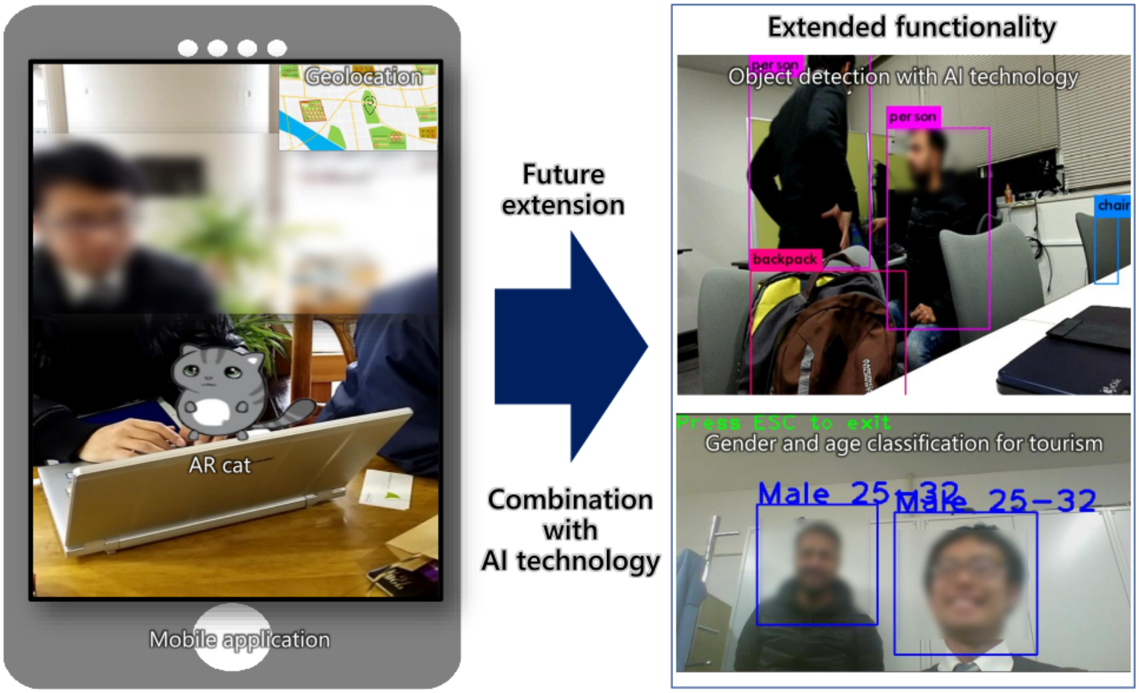


Figure A.1: Our prototype mobile application and future extension

mote tourism and expand IT Start-up companies and IT education development through programming schools to stimulate their local economy. When they recruit a game from a local school story after getting programming skills, they can implement a virtual entertainment park with plenty of natural resources.

We purchased embedded systems for education kits and AI sticks such as NVIDIA jetson tx2 and Intel Movidius stick in order to promote education for programming skills. We applied YOLO (You Only Look Once) [150], which is a kind of AI technology-based object recognition and localization. We can easily assign the AR object with this technology because it includes appearance feature-based recognition and localization. Figure 6 shows that our proposed prototype application and future extension of an example of object recognition and gender and age classification [151] using convolutional neural networks with embedded systems. Local people can expand AR mobile applications with this equipment because programming skills implemented these technologies. Programming skills can expand the future application.

Future onsite implementation

Through the above activity, we have plans for future onsite implementation combining this study's technology. Advanced mobile phone camera technology, which can capture images in hundreds of frames per second, can be used in disadvantaged areas such as rural communities with none or insufficient electric infrastructures. For example, rural areas have old structures such as houses and bridges to inspect manually for foundation stability. Although there are not many infrastructural engineers within these rural areas.In this situation, having a more straightforward method to inspect the stability of their foundations, infrastructure, and monitoring systems are very important for rural community members. Introducing an HFR video analysis that can extract pixel-level temporal frequency responses at hundreds of Hz or more, for structure, health inspection, and even monitoring, it would be a huge benefit to rural communities in Japan. It indicates that a cellular phone with an HFR video camera can function as a convenient mobile structure inspection system by implementing a cloud service-based software application for pixel-level temporary frequency analysis via the internet.

As we proposed a mobile application with geolocation, we can distribute efficient inspection through sightseeing activity using location map in some event. Additionally I would like to promote developing an automatic honeybee monitoring system using our proposed method in a disadvantaged area that has plenty of flower resources.

As we explained about CCD in chapter 5.1, CCD has become a severe social problem that causes significant economic losses to agricultural business because honeybees pollinate many crops. The number of honeybees is remarkably decreasing due to various issues such as enemy invasion and some disease, but the demand for honeybee is increasing in various fields such as the Agricultural field and medical field.

Although there is a significant demand for honeybee for management, it's challenging to monitor their activity and condition remotely. The younger generation and elderly people can also manage honeybee monitoring using automated honeybee activity moni-

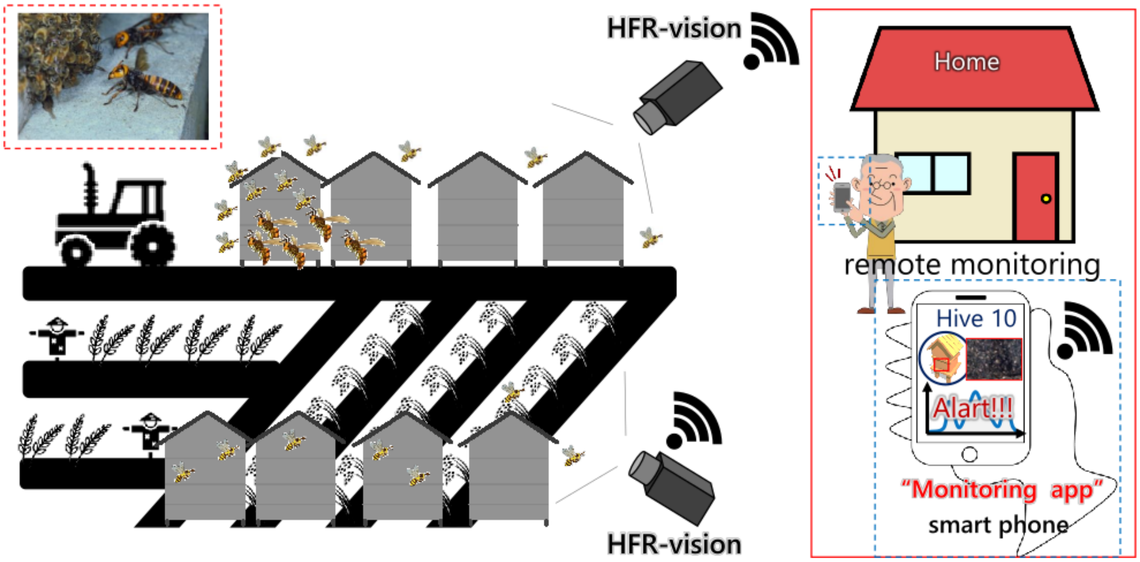


Figure A.2: Future onsite implementation for honeybee monitoring

toring technology. Figure 6 shows that our proposed system for honeybee monitoring can inform beekeepers using the mobile application when some incidents happen. When they can monitor which beehive is active or non-active and which beehive is attacked or not, they can manage the beehive more easily. If they utilize the honeybee monitoring system, they will make nutritious honey and their agricultural, economic, and health benefits more efficient. These plans have a sustainable possibility for onsite reverse innovation.

Bibliography

[1] Y. Lei, J. Lin, M.J. Zuo, and Z. He, “Condition monitoring and fault diagnosis of planetary gearboxes: A review,” *Measurement*, vol. 48, pp. 292–305, 2014.

[2] C. Li, R. V. Sanchez, G. Zurita, M. Cerrada, and D. Cabrera, “Fault diagnosis for rotating machinery using vibration measurement deep statistical feature learning,” *Sensors*, vol. 16, no. 6, pp. 895, 2016.

[3] M. Safizadeh, and S. Latifi, “Using multi-sensor data fusion for vibration fault diagnosis of rolling element bearings by accelerometer and load cell,” *Inform. Fusion*, vol. 18, pp. 1–8, 2014.

[4] A. Stanbridge, M. Martarelli, and D. Ewins, “Measuring area vibration mode shapes with acontinuous-scan LDV,” *Measurement*, vol. 35, pp. 181–189, 2004.

[5] J. Torano, R. Rodriguez, I. Diego, J. M. Rivas, and M.D. Casal, “FEM models including randomness and its application to the blasting vibrations prediction,” *Computers and Geotechnics*, vol. 33, no. 1, pp. 15–28, 2006.

[6] X. W. Ye, Y. Q. Ni, T. T. Wai, K. Y. Wong, X. M. Zhang, and F. Xu, “A vision-based system for dynamic displacement measurement of long-span bridges: algorithm and verification,” *Smart Struct. Syst.*, vol. 12, no. 3, pp. 363–379, 2013.

[7] D. Feng, and M. Q. Feng, “Vision-based multipoint displacement measurement for structural health monitoring,” *Struct. Control Health Monit.*, vol. 23, no. 5, pp. 876–890, 2015.

[8] E. Caetano, S. Silva, and J. Bateira, “A vision system for vibration monitoring of civil engineering structures,” *Exp. Tech.*, vol. 35, no. 4, pp. 74–82, 2011.

[9] G. Busca, A. Cigada, P. Mazzoleni, E. Zappa, “Vibration monitoring of multiple bridge points by means of a unique vision-based measuring system,” *Exp. Mech.*, vol. 54, pp. 255–271, 2014.

[10] X. Dai, X. Shao, Z. Geng, F. Yang, Y. Jiang, and X. He, “Vibration measurement based on electronic speckle pattern interferometry and radial basis function,” *Opt. Commun.*, vol. 355, pp. 33–43, 2015.

[11] I. Ishii, I. Ohara, T. Tatebe, and T. Takaki, “1000 fps target tracking using vibration-based image features,” *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 1837–1842, 2011.

[12] M. Jiang, T. Aoyama, T. Takaki, and I. Ishii. “Pixel-level and robust vibration source sensing in high-frame-rate video analysis,” *Sensors*, vol. 16, no.11, pp. 1842, 2016.

[13] M. Jiang, Q. Gu, T. Aoyama, T. Takaki, and I. Ishii. “Real-time vibration source tracking using high-speed vision,” *IEEE Sensors J.*, vol. 17, pp. 1513–1527, 2017.

[14] I. Ishii, T. Taniguchi, R. Sukenobe, and K. Yamamoto, “Development of high-speed and real-time vision platform, H3 Vision,” *IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pp. 3671–3678, 2009.

[15] I. Ishii, T. Tatebe, Q. Gu, Y. Moriue, T. Takaki, and K. Tajima, “2000 fps real-time vision system with high-frame-rate video recording,” *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 1536–1541, 2010.

[16] T. Yamazaki, H. Katayama, S. Uehara, A. Nose, M. Kobayashi, S. Shida, M. Odahara, K. Takamiya, Y. Hisamatsu, S. Matsumoto, L. Miyashita, Y. Watanabe, T. Izawa, Y. Muramatsu, and M. Ishikawa, “A 1 ms high-speed vision chip with 3D-

stacked 140GOPS column-parallel PEs for spatio-temporal image processing," *Abst. Int. Solid-State Circuits Conf.*, pp. 82–83, 2017.

[17] A. Sharma, K. Shimasaki, Q. Gu, J. Chen, T. Aoyama, T. Takaki, and I. Ishii, "Super high-speed vision platform that can process 1024×1024 images in real time at 12500 fps," *Proc. IEEE/SICE Int. Symp. Syst. Integr.*, pp. 544–549, 2016.

[18] D. Zhang, J. Guo, X. Lei, and C. Zhu, "A High-speed vision-based sensor for dynamic vibration analysis using fast motion extraction algorithms," *Sensors*, pp. 16–572, 2016.

[19] Q. Gu, T. Takaki, and I. Ishii, "A fast multi-object extraction algorithm based on cell-based connected components labeling," *IEICE Trans. Inform. Syst.*, vol. E95-D, no. 2, pp. 636–645, 2012.

[20] Q. Gu, T. Takaki, and I. Ishii, "Fast FPGA-based multi-object feature extraction," *IEEE Trans. Circ. Sys. Video Tech.*, vol. 23, no. 1, pp. 30–45, Jan. 2013.

[21] K. Okumura, K. Yokoyama, H. Oku, and M. Ishikawa, "1ms auto pan-tilt – video shooting technology for objects in motion based on saccade mirror with background subtraction," *Adv. Robot.*, vol. 29, no. 7, pp. 457–468, 2015.

[22] A. Namiki, K. Hashimoto, and M. Ishikawa, "A hierarchical control architecture for high-speed visual servoing," *Int. J. Robot. Res.*, vol. 22, no. 10–11, pp. 873–888, 2003.

[23] T. Senoo, A. Namiki, and M. Ishikawa, "Ball control in high-speed batting motion using hybrid trajectory generator," *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 1762–1767, 2006.

[24] A. Namiki and N. Ito, "Ball catching in Kendama game by estimating grasp condi-

tions based on a high-speed vision system and tactile sensors," *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, pp. 634–639, 2014.

[25] T. Aoyama, T. Takaki, T. Miura, Q. Gu, and I. Ishii, "Realization of flower stick rotation using robotic arm," *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pp. 5648–5653, 2015.

[26] L. Wang, R. X. Gao, "Condition monitoring and control for intelligent manufacturing," *Springer*, Berlin, 2006.

[27] E. Jantunen, "A summary of methods applied to tool condition monitoring in drilling," *Int. J. Mach. Tools Manuf.*, vol. 42, no. 9, pp. 997–1010, 2002.

[28] A. Z. Khan, A. B. Stanbridge, and D.J. Ewins, "Detecting damage in vibrating structures with a scanning LDV," *Opt. Lasers Eng.*, vol. 32, no. 6, pp. 583–592, 1999.

[29] D. Goyal, B. S. Pabla, "The Vibration Monitoring Methods and Signal Processing Techniques for Structural Health Monitoring: A Review," *Arch. Computat. Methods in Eng.* vol. 23, pp. 585–594, 2016.

[30] M. Dowell, and G. Sylvester, "Turbomachinery prognostics and health management via eddy current sensing: current developments," *Proc. IEEE Aerospace Conf.*, vol 3, pp 1–9, 1999.

[31] L. Maurice and M. L. Adams, "Rotating machinery vibration: from analysis to troubleshooting," *CRC Press*, 2000.

[32] C. P. Lawson and P. C. Ivey, "Tubomachinery blade vibration amplitude measurement through tip timing with capacitance tip clearance probes," *Sens. Actuators*, vol. 118, no. 1, pp. 14–24, 2005.

[33] R. G. Silva, R. L. Reuben, K. J. Baker, and S. J. Wilcox, "Tool wear monitoring of turning operations by neural network and expert system classification of a feature

set generated from multiple sensors," *Mech. Syst. Signal. Process.*, vol. 12, no. 2, pp. 319–332, 1998.

[34] Dimla DE, Lister PM "On-line metal cutting tool condition monitoring. I: force and vibration analyses." *Int. J. Mach. Tools Manuf.*, 2000

[35] Zhou, H., Yuan, Y. and Shi, C., "Object tracking using SIFT features and mean shift," *Comput. Vision and Image Understanding*, vol. 113, no. 3, pp. 345–352, 2009.

[36] S. Pavani, A. Delgado-Gomez, and A. Frangi, "Gaussian weak classifiers based on co-occurring haar-like features for face detection," *Pattern Anal. and Appl.*, vol. 17, no. 2, pp. 431–439, 2014.

[37] B. Wu, C. Kao, C. Jen, Y. Li, Y. Chen, and J. Juang, "A relative-discriminative-histogram-of-oriented-gradients-based particle filter approach to vehicle occlusion handling and tracking," *IEEE Trans. Ind. Electronics*, vol. 61, no. 8, pp. 4228–4237, 2014.

[38] A. Satpathy, X. Jiang, and H. L. Eng, "LBP-based edge-texture features for object recognition," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 1953–1964, 2014.

[39] S. Sivaraman, and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking and behavior analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, 2013.

[40] S. Rautaray, and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 1–54, 2015.

[41] I. Ishii, T. Taniguchi, K. Yamamoto, and T. Takaki, "High-frame-rate optical flow system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 1, pp. 105–112, 2012.

[42] I. Ishii, T. Tatebe, Q. Gu, and T. Takaki, “Color-histogram-based tracking at 2000 fps,” *J. Electron. Imaging*, vol. 21, no. 1, pp. 013010, 2012.

[43] Q. Gu, S. Raut, K. Okumura, T. Aoyama, T. Takaki, and I. Ishii, “Real-time image mosaicing system using a high-frame-rate video sequence,” *J. Robot. Mechatronics*, vol. 27, no. 1, pp. 12–23, 2015.

[44] I. Ishii, T. Ichida, Q. Gu, and T. Takaki. “500-fps face tracking system,” *J. Real-time Image Process.*, vol. 8, no. 4, pp. 379–388, 2013.

[45] S. Hertega, and J. Liljencrantz, “Measurement of human vocal fold vibrations with laser triangulation,” *Opt. Eng.*, vol. 40, no. 9, pp.2041–2044, 2001.

[46] J. H. Wu, R. S. Chang, and J. A. Jiang, “A novel pulse measurement system by using laser triangulation and a CMOS image sensor,” *Sensors*, vol. 7, no. 12, pp.3366–3385, 2007.

[47] P. Castellini, M. Martarelli, and E. P. Tomasini, “Laser doppler vibrometry: Development of advanced solutions answering to technology’s needs,” *Mech. System and Signal Process.*, vol. 20, no. 6, pp. 1265–1285, 2006.

[48] H. Oku, I. Ishii, and M. Ishikawa, “Tracking a protozoon using high-speed visual feedback,” *Proc. Int. IEEE-EMBS Conf. Microtechnologies in Medicine and Biology*, pp. 156–159, 2000.

[49] Q. Gu, T. Kawahara, T. Aoyama, T. Takaki, I. Ishii, A. Takemoto, and N. Sakamoto, “LOC-based high-throughput cell morphology analysis system,” *IEEE Trans. Automat. Sci. Eng.*, vol. 12, no. 4, pp. 1346–1356, 2015.

[50] H. Yang, Q. Gu, T. Aoyama, T. Takaki, and I. Ishii, “Dynamics-based stereo visual inspection using multidimensional modal analysis,” *IEEE Sensors Journal*, vol. 13, no. 12, pp. 4831–4843, 2013.

[51] S. Argentieri, P. Danes, and P. Soueres, "A survey on sound source localization in robotics: Binaural to array processing methods," *Comput. Speech Lang.*, vol. 34, no. 1, pp. 87–112, 2015.

[52] Y. C. Lu and M. Cooke, "Binaural estimation of sound source distance via the direct-to-reverberant energy ratio for static and moving sources," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 7, pp. 1793–1805, 2010.

[53] U. H. Kim, K. Nakadai, and H. G. Okuno, "Improved sound source localization in horizontal plane for binaural robot audition," *Appl. Intell.*, vol. 42, no. 1, pp. 63–74, 2015.

[54] X. Zhong, W. Yost, and L. Sun, "Dynamic binaural sound source localization with ITD cues: Human listeners," *J. Acoust. Soc. Amer.*, vol. 137, no. 4, pp. 2376–2376, 2015.

[55] M. Raspaud, H. Viste, and G. Evangelista, "Binaural source localization by joint estimation of ILD and ITD," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 1, pp. 68–77, 2010.

[56] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 24, no. 4, pp. 320–327, 1976.

[57] J. Benesty, "Adaptive eigenvalue decomposition algorithm for passive acoustic source localization," *J. Acoust. Soc. Amer.*, vol. 107, no. 1, pp. 384–391, 2000.

[58] J. Chen, J. Benesty, and Y. Huang, "Time delay estimation in room acoustic environments: an overview," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–19, 2006.

[59] R. O. Schmidt, “Multiple emitter location and signal parameter estimation,” *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, 1986.

[60] K. Okutani, T. Yoshida, K. Nakamura, and K. Nakadai, “Outdoor auditory scene analysis using a moving microphone array embedded in a quadrocopter,” *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pp. 3288–3293, 2012.

[61] L. J. Griffiths and C. W. Jim, “An alternative approach to linearly constrained adaptive beamforming,” *IEEE Trans. Antennas Propag.*, vol. 30, no. 1, pp. 27–34, 1982.

[62] J. M. Valin, F. Michaud, and J. Rouat, “Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering,” *Robot. Auton. Syst.*, vol. 55, no. 3, pp. 216–228, 2007.

[63] F. Ribeiro, D. Florencio, D. Ba, and C. Zhang, “Geometrically constrained room modeling with compact microphone arrays,” *IEEE Trans. Audio Speech Lang. Process.*, vol. 20, no. 5, pp. 1449–1460, 2012.

[64] F. Keyrouz, “Advanced binaural sound localization in 3-D for humanoid robots,” *IEEE Trans. Instrum. Meas.*, vol. 63, no. 9, pp. 2098–2107, 2014.

[65] R. Li, and D. He, “Rotational machine health monitoring and fault detection using EMD-based acoustic emission feature quantification,” *IEEE Trans. Instrum. Meas.*, vol. 61, no. 4, pp. 990–1001, 2012.

[66] J. A. Ballesteros, E. Sarradj, M. D. Fernandez, T. Geyer, and M. J. Ballesteros, “Noise source identification with beamforming in the pass-by of a car,” *Appl. Acoust.*, vol. 93, pp. 106–119, 2015.

[67] P. Castellini, A. Sassaroli, A. Paonessa, A. Peiffer, and A. Roeder, “Average beamforming in reverberant fields: Application on helicopter and airplane cockpits,” *Appl. Acoust.*, vol. 74, no. 1, pp. 198–210, 2013.

[68] K. Nakadai, T. Takahashi, H. G. Okuno, H. Nakajima, Y. Hasegawa, and H. Tsujino, “Design and implementatiaon of robot audition system ’ HARK ’ - open source software for listening to three simultaneous speakers,” *Adv. Robot.*, vol. 24, no. 5-6, pp. 739–761, 2010.

[69] D. Yang, Z. Wang, B. Li, and X. Lian, “Development and calibration of acoustic video camera system for moving vehicles,” *J. Sound and Vib.*, vol. 330, no. 11, pp. 2457–2469, 2011.

[70] B. Barsikow, W. F. King, and E. Pfizenmaier, “Wheel/rail noise generated by a high-speed train investigated with a line array of microphones,” *J. Sound and Vib.*, vol. 118, no. 1, pp. 99–122, 1987.

[71] Y. Li, M. Smith, and X. Zhang, “Measurement and control of aircraft landing gear broadband noise,” *Aerosp. Sci. Technol.*, vol. 23, no. 1, pp. 213–223, 2012.

[72] T. Pham, and N. Srour, “TTCP AG-6: acoustic detection and tracking of UAVs,” *Defense and Secur.*, pp. 24–30, 2004.

[73] J. Busset, F. Perrodin, P. Wellig, B. Ott, K. Heutschi, T. Ruhl, and T. Nussbaumer, “Detection and tracking of drones using advanced acoustic cameras,” *SPIE Secur. + Defence*, pp. 96470F–96470F, 2015.

[74] Dedrone, “Multi-sensor drone warning system.” San francisco, USA, (2020). Accessed: June. 15, 2020. [Online]. Available: http://www.dedrone.com/en/dronetracker/drone-detection-hardware.

[75] Droneshield, “How droneshield works.” Sydney, Australia, (2020). Accessed: June. 15, 2020. [Online]. Available: https://www.droneshield.com/how-droneshield-works1.

[76] K. Shimasaki, T. Okamura, M. Jiang, T. Takaki, I. Ishii, and K. Yamamoto, “HFR-video-based image pattern recognition using pixel-level temporal frequency response matching,” *Proc. IEEE Int. Conf. Automat. Sci. Eng.*, pp. 451–456, 2018.

[77] J. Seo, H. Ma, and T. K. Saha, “A joint vibration and arcing measurement system for online condition monitoring of on-load tap changer of power transformer,” *IEEE Trans. Power Del.*, vol. 32, no. 2, pp. 1031–1038, 2017.

[78] Y. Park, M. Jeong, S. B. Lee, J. A. Antonino-Daviu, and M. Teska, “Influence of blade pass frequency vibrations on MCSA-based rotor fault detection of induction motors,” *IEEE T. Ind. Appl.*, vol. 53, no. 3, pp. 2049–2058, 2017.

[79] W. Pan, X. Kong, and J. Xu, “Measurement and analysis system of vibration for the detection of insect acoustic signals,” *2016 Asia-Pacific Int. Symp. Electromagn. Compat. (APEMC)*, 2016.

[80] N. S. Marne, M. S. Nagmode, and R. D. Komati, “Vibration measurement system with accelerometer sensor based on ARM,” *Int. J. Emerg. Technol. Adv. Eng.*, vol. 4, no. 4, 2014.

[81] A. Devillez and D. Daniel, “Tool vibration detection with eddy current sensors in machining process and computation of stability lobes using fuzzy classifiers, *Mech. Sys. Signal Pr.*, vol. 21, no. 1, pp. 441–456, 2007.

[82] Y. R. Garcia, J. M. Corres, and J. Goicoechea, “Vibration detection using optical fiber sensors,” *J. Sens.*, vol. 2010, pp. 1–12 2010.

[83] F. Garoi, P. C. Logofatu, D. Apostol, C. Udrea, and P. Schiopu, “Interferometric Vibration displacement measurement,” *Physics*, vol. 62, pp. 671–677, 2010.

[84] D. Mas, J. Espinosa, J. Perez, C. Illueca, B. Ferrer, and A. B. Roig, “High speed

imaging and algorithms for non-invasive vibrations measurement," *Proc. EVACES 2011 Exp. Vib. Anal. Civ. Eng. Struct.*, pp. 531–538, 2011.

[85] E. Carrera and A. Pagani, "Free vibration analysis of civil engineering structures by component-wise models," *J. Sound Vib.*, vol. 333, no. 19, pp. 4597–4620, 2014.

[86] S. Wan, D. Li, J. Gao, R. Roy, and Y. Tong, "Process and knowledge management in a collaborative maintenance planning system for high value machine tools," *Comput. Ind.*, vol. 84, pp. 14–24, 2017.

[87] Y. Nie, I. Ishii, K. Yamamoto, K. Orito, and H. Matsuda, "Real-time scratching behavior quantification system for laboratory mice using high-speed vision," *J. Real-Time Image Process.*, vol. 4, pp. 181–190, 2009.

[88] S. Sakuma, K. Kuroda, C. Tsai, W. Fukui, F. Arai, and M. Kaneko, "Red blood cell fatigue evaluation based on the close-encountering point between extensibility and recoverability," *Lab on a Chip*, vol. 14, pp. 1135–1141, 2014.

[89] Q. Gu, T. Aoyama, T. Takaki, and I. Ishii, "Simultaneous vision-based shape and motion analysis of cells fast-flowing in a microchannel," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, pp. 204–215, 2015.

[90] H. Yang, T. Takaki, and I. Ishii, "Simultaneous Dynamics-Based Visual Inspection Using Modal Parameter Estimation," *J. Robot. Mech.*, vol. 23, no. 1, 2011.

[91] T. Aoyama, L. Li, M. Jiang, K. Inoue, T. Takaki, I. Ishii, H. Yang, C. Umemoto, H. Matsuda, M. Chikaraishi, and A. Fujiwara, "Vibration sensing of a bridge model using a multithread active vision system," *IEEE/ASME Trans. Mechatronics*, vol. 23, no 1, pp. 179–189, 2017.

[92] "Anti-drone market by technology (laser, kinetic, and electronics), application (de-

tection and detection & disruption), vertical (military & defense, homeland security, and commercial), and geography - global forecast to 2023," 2017,

[93] A. Yilmaz, O. Javed, and M. Shah."Object tracking: a survey," *ACM Comput. Surv.*, vol. 38, no. 4, Jan. 2006.

[94] A. Rozantsev, V. Lepetit, and P. Fua, "Detecting flying objects using a single moving camera," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 39, no. 5, pp. 879–892, 2017.

[95] R. Chandra, R., Menon, L., Dagum, D., Kohr, D., Maydan, and J. McDonald, "Parallel programming in openMP," *Morgan Kaufmann Publishers*, San Francisco, CA., 2000.

[96] G. B. Patricio-Roberto and M. J. O. Campos, "Aspects of landscape and pollinators-what is important to bee conservation?," *Diversity*, vol. 6, pp. 158–175, 2014.

[97] G. R. Williams, D. R. Tarry, D. van Engelsdorp, M. P. Chauzat, D. L. Cox-Foster, K. S. Delaplane, P. Neumann, J. S. Pettis, R. E. L. Rogers, and D. Shutler, "Colony collapse disorder in context," *BioEssays*, vol. 32, no. 10, pp. 845–846, 2010.

[98] D. van Engelsdorp, J. D. Evans, C. Saegerman, C. Mullin, E. Haubruge, B. K. Nguyen, M. Frazier, J. Frazier, D. Cox-Foster, Y. Chen, R. Underwood, D. R. Tarpy, and J. S. Pettis, "Colony collapse disorder: A descriptive study," *PLoS One*, vol. 4, no. 8, pp. 1–17, 2009.

[99] D. van Engelsdorp, D. Caron, J. Hayes, R. Underwood, M. Henson, A. Spleen, M. Andree, R. Snyder, K. Lee, K. Roccasecca, M. Wilson, J. Wilkes, E. Lengerich, and J. Pettis "A national survey of managed honey bee 2010-11 winter colony losses in the USA: results from the bee informed partnership," *J. Apic. Res.*, vol. 51, pp. 115–124, 2012.

[100] D. Goulson, E. Nicholls, C. Botias, and E. L. Rotheray, “Bee declines driven by combined stress from parasites, pesticides and lack of flowers,” *Science*, vol. 347, no. 6229, pp. 1255957, 2015.

[101] P. Neumann and N. L. Carreck, “Honey bee colony losses,” *J. Apic. Res.*, vol. 49, pp. 1–6, 2010.

[102] B. N. Gates, ”The temperature of the bee colony,” *Bull. US Dept. Agric.*, vol. 96, pp. 1–29, 1914.

[103] W. G. Meikle and N. Holst, “Application of continuous monitoring of honeybee colonies,” *Apidologie*, vol. 46, no. 1, pp. 10–22, 2015.

[104] W. G. Meikle, N. Holst, G. Mercadier, F. Derouane, and R. R. James, “Using balances linked to dataloggers to monitor honeybee colonies,” *J. Apic. Res.*, vol. 45, no. 1, pp. 39–41, 2006.

[105] W. G. Meikle, B. G. Rector, G. Mercadier, and N. Holst, “Within-day variation in continuous hive weight data as a measure of honey bee colony activity,” *Apidologie*, vol. 39, no. 6, pp. 694–707, Dec. 2008.

[106] S. C. Thoenes and S. L. Buchmann, “Colony abandonment by adult honey bees: A behavioral response to high tracheal mite infestation?,” *J. Apic. Res.* vol. 31, no. 3–4, pp. 167–168, 1992.

[107] E. E. Southwick and J. N. Mugaas, “A hypothetical homeotherm: The honey bee hive,” *Comp. Biochem. Physiol.* vol. 40, no. 4, pp. 935–944, 1971.

[108] K. V. Nerum and H. Buelens, “Hypoxiacontrolled winter metabolism in honeybees (Apis mellifera),” *Comp. Biochem. Physiol.* vol. 117, no. 4, pp. 445–455, 1997.

[109] F. Kronenberg and H. C. Heller, “Colonial thermoregulation in honey bees (Apis mellifera),” *J. Comp. Physiol.* vol. 148, no. 1, pp. 65–76, 1982.

[110] H. Human, S. W. Nicolson, and V. Dietemann, "Do honeybees," *Apis mellifera scutellata*, vol. 93, no. 8, pp. 397–401, 2006.

[111] K. V. Frisch. "The dance language and orientation of bees," *Harvard University Press*, Cambridge, Massachusetts, 1967.

[112] L. Fahrenholz, I. Lamprecht, and B. Schricker, "Calorimetric investigations of the different castes of honey bees, apis mellifera carnica," *J. Comp. Physiol. B*, vol. 162, no. 2, pp. 119–130, 1992.

[113] H. A. Toufailia, M. Couvillon, F. Ratnieks, and C. Grüter, "Honey bee waggle dance communication: Signal meaning and signal noise affect dance follower behaviour," *Behav. Ecol. Sociobiol.*, vol. 67, no. 4, pp. 549–556, 2013.

[114] M. Hrncir, F. G. Barth, and J. Tautz, "Vibratory and airborne-sound signals in bee communication (Hymenoptera)," *Insect Sound Commun. Physiol. Behav. Ecol. Evol.*, pp. 421, 2005.

[115] J. C. Nieh and J. Tautz, "Behaviour-locked signal analysis reveals weak 200-300 Hz comb vibrations during the honey bee waggle dance," *J. Exp. Biol.* vol. 203, no. 10, pp. 1573–1579, 2000.

[116] M. Bencsik, J. Bencsik, M. Baxter, A, Lucian, J. Romieu, and M. Millet, "Identification of the honey bee swarming process by analysing the time course of hive vibrations," *Comput. Electron. Agric.* vol. 76, no. 1, pp. 44–50, 2011.

[117] S. Ferrari, M. Silva, M., Guarino, and D. Berckmans, "Monitoring of swarming sounds in bee hives for early detection of the swarming period," *Comput. Electron. Agric.* vol. 64, no. 1, pp. 72–77, 2008.

[118] D. A. Mezquida and J. L. Martinez, "Platform for bee-hives monitoring based on

sound analysis. A perpetual warehouse for swarm's daily activity span," *J. Agric. Res.*, vol. 7, no. 4, pp. 824–828, 2009.

[119] A. R, McLellan, "Honey bee colony weight as an index of honey production and nectar flow: a critical evaluation," *J. Appl. Ecol.*, vol. 14, no. 2, pp. 401–408, 1997.

[120] M. H. Pham-Delegue, A. Decourtye, L. Kaiser, and J. Devillers, "Behavioural methods to assess the effects of pesticides on honey bees," *Apidologie* vol. 33, no. 5, pp. 425–432, 2002.

[121] A. C. Faberge, "Apparatus for recording the number of bees leaving and entering a hive," *J. Sci. Instr.*, vol. 20, no. 2, pp. 28–311, 1943.

[122] H. G. Spangler, "Photoelectrical counting of outgoing and incoming honey-bees," *J. Econ. Entomol.*, vol. 62, no. 5, pp. 1183–1184, 1969.

[123] R. M. Burrill and A. Dietz, "The response of honey bees to variations in solar radiation and temperature," *Apidologie*, vol. 12, no. 4, pp. 319–328, 1981.

[124] C. Liu, J. Leonard, and J. J. Feddes, "Automated monitoring of flight activity at a beehive entrance using infrared light sensors," *J. Apic. Res.*, vol. 29, no. 1, pp. 20–27, 1990.

[125] R. G. Danka and L. D. Beaman, "Flight activity of USDA-ARS Russian honey bees (Hymenoptera: Apidae) during pollination of lowbush blueberries in Maine," *J. Econ. Entomol.*, vol. 100, no. 2, pp. 267–272, 2007.

[126] S. S. Schneider, J. A. Stamps, and N. E. Gary, "The vibration dance of the honey bee. I. Communication regulating foraging on two time scales," *Anim. Behav.*, vol. 34, no. 2, pp. 377–385, 1986.

[127] S. Streit, F. Bock, C. W. W. Pirk, and J. Tautz, “Automatic life-long monitoring of individual insect behaviour now possible,” *Zoology*, vol. 106, no. 3, pp. 169–171, 2003.

[128] J. A. Shaw, N. L. Seldomridge, D. L. Dunkle, P. W. Nugent, L. H. Spangler, J. J. Bromenshenk, C. B. Henderson, J. H. Churnside, and J. J. Wilson, “Polarization lidar measurements of honey bees in flight for locating land mines,” *Opt. Express.*, vol. 13, no. 15, pp. 5853–5863, 2005.

[129] D. S. Hoffman, A. R. Nehrir, K. S. Repasky, J. A. Shaw, and J. L. Carlsten, “Range-resolved optical detection of honeybees by use of wing-beat modulation of scattered light for locating land mines,” *Appl. Opt.*, vol. 46, no. 15, pp. 3007–3012, 2007.

[130] E. S. Carlsten, G. R. Wicks, K. S. Repasky, J. K. Carlsten, J. J. Bromenshenk, and C. B. Henderson, “Field demonstration of a scanning lidar and detection algorithm for spatially mapping honeybees for biological detection of land mines,” *Appl. Opt.*, vol. 50, no. 14, pp. 2112-2123, 2011.

[131] V. Estivill-Castro, D. C. Lattin, F. Suraweera, and V. Vithanage, “Tracking bees – a 3D, outdoor small object environment,” *Proc. Int. Conf. Image Process.*, 2003.

[132] T. Landgraf and R. Rojas. “Tracking honey bee dances from sparse optical flow fields,” *Tech. Rep. B 07-11*, Freie Universitat Berlin, 2007.

[133] J. M. Campbell, D. C. Dahn, and D. A. J. Ryan, “Capacitance-based sensor for monitoring bees passing through a tunnel,” *Meas. Science Technol.*, vol. 16, no. 12, pp. 2503, 2005.

[134] J. Santoyo, W. Azarcoya, M. Valencia, A. Torres, and J. Salas, “Frequency analysis of a bumblebee (bombus impatiens) wingbeat,” *Patt. Anal. Appl.*, vol. 19, no. 2, pp.487–493, 2016.

[135] A. Decourtye, J. Devillers, P. Aupinel, F. Brun, C. Bagnis, J. Fourrier, and M. Gauthier, “Honeybee tracking with microchips: a new methodology to measure the effects of pesticides,” *Ecotoxicology*, vol. 20, no. 2, pp. 429–437, 2011.

[136] J. Campbell, L. Mummert, and R. Sukthanker, “Video monitoring of honey bee colonies at the hive entrance,” *Proc. Workshop Visual Observ. Anal. Anim. Insect Behav.*, vol. 7, 2008.

[137] T. Kimura, M. Ohashi, K. Crailsheim, T. Schmickl, R. Okada, G. Radspieler, and H. Ikeno, “Development of a new method to track multiple honey bees with complex behaviors on a flat laboratory arena,” *PLoS One*, vol. 9, no. 1, pp. 1–12, 2014.

[138] D. J. Kale, R. Tashakkori, and R. M. Parry, “Automated beehive surveillance using computer vision,” *Proc. IEEE SoutheastCon*, 2015.

[139] G. J. Tu, M. K. Hansen, P. Kryger, and P. Ahrendt, “Automatic behaviour analysis system for honeybees using computer vision,” *Comput. Electron. Agric.*, vol. 122, pp. 10–18, 2016.

[140] G. Chiron, P. Gomez-Kramer, and M. Menard, “Detecting and tracking honeybees in 3D at the beehive entrance using stereo vision,” *EURASIP J. Image Video Process.*, vol. 1, pp. 1–17, 2013.

[141] E. Caetano, S. Silva, and J. Bateira, “A vision system for vibration monitoring of civil engineering structures,” *Exp. Techn.*, vol. 35, no. 4, pp. 74–82, 2011.

[142] H. G. Maas and U. Hampel, “Photogrammetric techniques in civil engineering material testing and structure monitoring,” *Photogram. Eng. Remote Sens.*, vol. 72, no. 1, pp. 39–45, 2006.

[143] J. G. Chen, N. Wadhwa, F. Durand, W. T. Freeman, and O. Buyukozturk, “De-

velopments with motion magnification for structural modal identification through camera video," *Dyn. Civil Struct.*, vol. 2, pp. 49–57, 2015.

[144] J. Lohscheller, U. Eysholdt, H. Toy, and H. Dollinger, "Phonovibrography: Mapping high-speed movies of vocal fold vibrations into 2-D diagrams for visualizing and analyzing the underlying laryngeal dynamics," *IEEE Trans. Med. Imaging*, vol. 27, no. 3, pp. 300–309, 2008.

[145] D. D. Mehta, D. D. Deliyski, T. F. Quatieri, and R. E. Hillman, "Automated measurement of vocal fold vibratory asymmetry from high-speed videoendoscopy recordings," *J. Speech Lang. Hear. Res.* vol. 54, no. 1, pp. 47–54, 2011.

[146] A. P. Pinheiro, D. E. Stewart, C. D. Maciel, J. C. Pereira, and S. Oliveira, "Analysis of nonlinear dynamics of vocal folds using highspeed video observation and biomechanical modeling," *Digit. Signal Process.*, vol. 22, no. 2, pp. 304–313, 2012.

[147] K. Shimasaki, M. Jiang, T. Takaki, I. Ishii, and K. Yamamoto, "HFR-video-based honeybee activity sensing using pixel-level short-time Fourier transform," *Proc. IEEE Sensors 2018*, pp. 812–815, 2018.

[148] K. S. Repasky, J. A. Shaw, R. Scheppele, C. Melton, J. L. Carsten, and L. H. Spangler, "Optical detection of honeybees by use of wing-beat modulation of scattered laser light for locating explosives and land mines," *Appl. Opt.*, vol. 45, no. 8, pp. 1839–1843, 2006.

[149] A. M. Wenner, "The flight speed of honeybees: a quantitative approach," *J. Apic. Res.*, vol. 2, pp. 25–32, 1963.

[150] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," *IEEE Int. Conf. Comput. Vision and Pattern Recognit.*, pp. 779–788, 2016.

[151] G. Levi, and T. Hassner, “Age and gender classification using convolutional neural networks,” *IEEE Int. Conf. Comput. Vision and Pattern Recognit.*, pp. 34–42, 2015.

Acknowledgment

Firstly, I would like to thank my advisor, **Prof. Idaku Ishii**, who gave me an opportunity to join the robotics engineering laboratory and has guided me through these six years. He has taught me lot of skills and attitudes for academic research in aspects of a new research consideration, a research process method, a research summarization such as presentations and papers. I want to express my gratitude again to **Prof. Idaku Ishii**.

Additionally, I would like to express my gratitude to **Dr. Takeshi Takaki**, **Dr. Qingyi Gu**, **Dr. Taku Senoo**, **Dr. Mingjun Jiang**, **Dr. Shaopeng Hu**, **Dr. Sushil Raut**, and **Dr. Michiaki Inoue**. They helped me overcome the unfamiliarity with a new experimental environment when I joined our laboratory. I would like to express my heartfelt gratitude to **Ms. Yukari Kaneyuki**, and **Ms. Michiko Kanzaki** (educational administrator), and **Ms. Rumi Horiuchi** and **Ms. Arisa Tomura** (laboratory secretary). They were my most reliable staff in our institution; I received thoughtful attention both in my study. I want to express my sincere thanks to the bachelor, master, and doctoral students in Robotics Laboratory for their help in my life and research. I am grateful to the teachers, clerical staff, and doctoral students, especially **Ms. Diana Khan** in the TAOYAKA Program. They had the opportunity to experience many fieldwork and events that general doctoral students do not have as well. I had come across valuable experience through onsite course rotation when I was a student in the TAOYAKA program. I will just subscribe to my point for those experiences regarding onsite projects and training from points of the cultural, social, and technical aspects.

Finally, I want to express my profound gratitude to my family for their support and warm encouragement until now throughout my life.

July, 2020

Kohei Shimasaki