Detecting and tracking honeybees in 3D at the beehive entrance using stereo vision

Guillaume Chiron, Petra Gomez-Krämer, Ménard Michel

Detecting and tracking honeybees in 3D at the beehive entrance using stereo vision

Guillaume Chiron[1*]
[*]Corresponding author
Email: guillaume.chiron@univ-lr.fr

Petra Gomez-Krämer[1]
Email: petra.gomez@univ-lr.fr

Michel Ménard[1]
Email: michel.menard@univ-lr.fr

[1]L3i, University of La Rochelle, La Rochelle 17000, France

Abstract

In response to recent needs of biologists, we lay the foundations for a real-time stereo vision-based system for monitoring flying honeybees in three dimensions at the beehive entrance. Tracking bees is a challenging task as they are numerous, small, and fast-moving targets with chaotic motion. Contrary to current state-of-the-art approaches, we propose to tackle the problem in 3D space. We present a stereo vision-based system that is able to detect bees at the beehive entrance and is sufficiently reliable for tracking. Furthermore, we propose a detect-before-track approach that employs two innovating methods: hybrid segmentation using both intensity and depth images, and tuned 3D multi-target tracking based on the Kalman filter and Global Nearest Neighbor. Tests on robust ground truths for segmentation and tracking have shown that our segmentation and tracking methods clearly outperform standard 2D approaches.

Keywords

Stereo vision; RGB-D segmentation; 3D multi-target tracking; Honeybee; Beehive monitoring

1 Introduction

There is currently much debate about pesticides and risk assessment procedures. It has been demonstrated that the cumulative effect of pesticides, even at doses below the detectable threshold, weakens bees and causes significant mortality in the colony [1]. Indeed, beekeepers regularly observe bees with abnormal behaviors at the beehive entrance. As far as we know, no quantification or qualification of these abnormal behaviors has been attempted, possibly due to technical difficulties. In this context, it has become urgent to develop new approaches for phytosanitary product evaluation based on measurable indicators [2].

Thus, in order to meet the needs of biologists, it is essential to collect data on bees at different levels: numbers of bees, trajectories, and behaviors. When done manually with videos, the process is time-consuming and suffers from a lack of precision due to human error. We believe that computer vision can effectively achieve these tasks with accuracy.

This work was performed in the context of the environment monitoring platform schematized in Figure 1. The main purpose of this platform is to compare and contrast environmental data and to make them available to biologists. A bee counter based on computer vision was introduced in [3]; it was the first module to be included in the platform. The system described in this current paper corresponds to a module that is under development. Section 6 indicates potential extensions to the platform that could be implemented in the future such as a behavior analysis module.

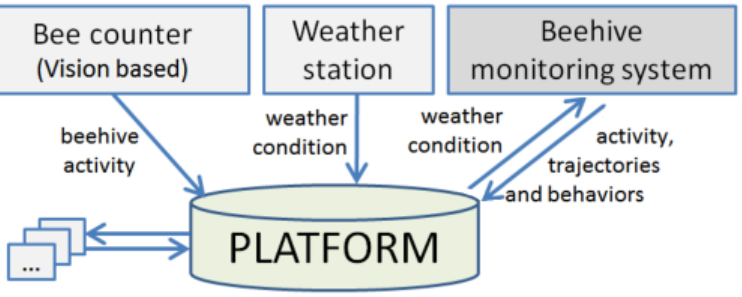

Figure 1: Environmental monitoring platform. Environmental monitoring platform sharing information with different modules. This ongoing work is undertaken in the context of the beehive monitoring system module currently under development.

Monitoring bees automatically in an uncontrolled outdoor environment involves a lot of constraints. Working in completely natural conditions raises problems such as sudden changes in light and background soiling. However, the main problem is the nature of the target. Bees are small and fast-moving targets, and their motion may be chaotic. There is often a lot of activity around the flight board, in front of the beehive, which results in a lot of occlusions. Given the real-time acquisition constraint of our application regarding the amount of data that has to be processed, a simple and efficient approach was required.

1.1 Related work

Automated honeybee counters were the first technically feasible application to be introduced. Over the last 40 years, different approaches have been explored: mechanical counters [4], less intrusive infrared sensors [5], individual identification by radio frequency identification [6], and more recently, video-based counters with [7] and without identification [3]. In [7], tagged honeybees were detected using a circular Hough transform, and tag characters were recognized using a support vector machine.

The development of increasingly powerful computers has led to a growing interest among biologists in applications based on computer vision. The papers discussed below used trajectometry based on videos. The whole tracking process can be split into three parts which are presented separately below: segmentation, tracking, and behavior analysis.

1.1.1 Detection

For target detection, several methods have been proposed. Detect-before-track approaches are generally based on a segmentation process. Potential targets are associated with existing or new tracks using an assignment method. Many studies based on this approach used a background subtraction of varying sophistication (e.g., [8-10]). In [9], potential false alarms were filtered using a shape (ellipsis-based) matching process. In contrast, other methods do not require any background subtraction. The authors of [11] detected bees using the well-known Viola-Jones method [12], and the authors of [13] introduced

an approach based on vector quantization, which is able to detect individual bees among hundreds of walking bees. In [14], flying bats were detected taking advantage of multiple cameras by directly applying a direct linear transform. In the case of track-before-detect approaches, the position of each target is first estimated, and the probability that the estimation corresponds to a potential target drives the next estimation. For this kind of approach, a likelihood function based on appearance models (e.g., a precomputed 'eigenbee' in [15] or an adaptive appearance model in [16,17]) is used.

1.1.2 Tracking

Many methods have been proposed for tracking. When following only a clearly detected target moving along a simple trajectory, approaches such as those used in [10] and [18] (nearest neighbor or mean shift) may be sufficient. However, tracking multiple targets involves assignment problems because of miss detections and false alarms. The authors of [8,9] used Global Nearest Neighbor (GNN) for track assignment, instantiation, and destruction. In [14], the authors tracked multiple flying targets using a Multiple Hypothesis Tracker (MHT). In contrast to GNN, MHT considers different hypotheses for the assignment decision process. In [15-17,19], a non-linear motion model was considered for their targets and they based their tracking on a particle filter [20], which corresponds to a track-before-detect approach. In [19], the authors introduced an MRF-augmented particle filter for multiple targets at a reasonable computational cost. This contrasts with the three other methods used in [15-17] which are less suitable when working on multiple interacting targets.

1.1.3 Behavior

Few studies have looked at behavior analysis based on trajectories. In [21], supervised k-means clustering was used to detect low-level motion patterns from tracks, and then a hidden Markov model was employed to identify high-level behaviors from the patterns. The authors of [16] and [9] went further, improving tracking by adapting the motion model driven by the detected behavior.

The results presented in different papers are difficult to interpret because no common ground truth was used to make objective comparisons. Also, significant parameters such as the observation frequency or the lighting conditions vary considerably from one application to another. Table 1 summarizes the applications and their characteristics. The Loc/obs column indicates whether the targets move on a plane (2D) or in space (3D) and whether tracking recovers the trajectory in 2D or 3D. The Cam/fps column indicates the number of cameras used simultaneously and the frequency of acquisition. Finally, Table 1 shows that when dealing with many flying targets in natural conditions, there is only a limited range of applications, suggesting that new methods should be explored.

Table 1: Papers related to insect tracking

Reference	Targets	Loc/obs	Cam/fps	Detection/likehood method	Tracking method
[8]	<15 ants	2D/2D	1/30	ABS	Basic GNN
[19]	<20 ants	2D/2D	1/30	Appearance model	MRF-augmented FP
[15]	1 bee*	2D/2D	1/15	Eigenbee (PCA)	PF
[16]	1 bee*	2D/2D	1/15	Adaptive appearance model (e.g., color image)	PF supported by a behavior model
[17]	1 bee*	2D/2D	1/15	Weighted adaptive appearance model (e.g., color image)	Idem + geometric constraints
[13]	<100 bees	2D/2D	1/30	Vector quantization (VQ)	Overlapping ellipse
[18]	<100 bees	2D/2D	1/14	Tag detection (method not mentioned)	Mean shift
[11]	1 bee	2D/2D	1/?	Viola-Jones detector	Combined NN classification of BOF
[9]	n bees	3D/2D	1/30	ABS + ellipse matching	Basic GNN
[10]	1 bee	3D/2D	1/24	ABS	NN
[14]	<100 bats	3D/3D	3 IR/125	Direct linear transform	MHT
Our	<25 bees	3D/3D	2/60	Hybrid intensity/depth segmentation	GNN and 3D (re)projection

Summary of recent papers related to insect and animal tracking. NN, near neighbor; BOF, bag of feature; ABS, adaptive background subtraction; PF, particle filter; KF, Kalman filter; GNN, Global Near Neighbor; IR, infrared; PCA, principal component analysis; MHT, multi-hypothesis tracker. *Not explicitly mentioned in the paper but possibly extensible to several targets.

1.2 Contributions

This paper contributes to the field by demonstrating the feasibility of the system. Section 1.1 clearly summarizes the insect detection and tracking applications that have been developed in recent years. It would appear that stereo vision has never been used for close-range tracking of small flying targets. As a first and general contribution, we demonstrate that honeybees in flight can be tracked in 3D using a stereo vision camera, and we describe in detail a suitable acquisition system for our application (illustrated in Figure 2).

Figure 2: Stereo vision camera. A stereo vision camera capturing the entry of a beehive including the entire flight board.

Moreover, we provide a complete detect-before-track chain suitable for small and fast flying targets in

3D. Our contribution here is twofold. First, we propose a hybrid segmentation method based on both depth and intensity images that works in completely natural conditions. The evaluation of the segmentation method is shown in Section 5.1 and relies on an accurate and robust ground truth. Secondly, we introduce a robust real-time 3D tracking method based on the Kalman filter for motion estimation and GNN for observation in order to track assignments. The latter contribution focuses on a reprojection mechanism, which allows the tracker to maintain a track in 3D with only 2D observations during a temporary period before new 3D observations are received. As an example, Figure 3 shows tracks recovered by our detect-before-track system.

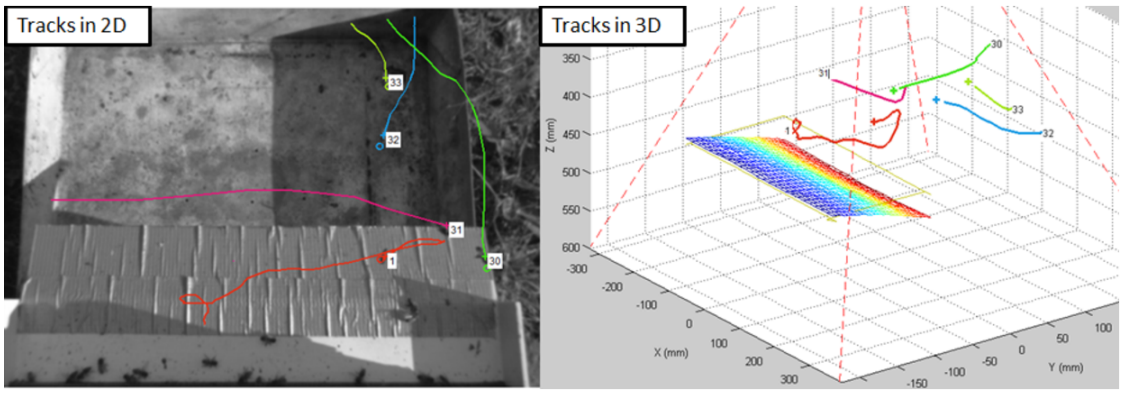

Figure 3: Example of bee tracks. Example of bee tracks shown on a 2D top view (left image) and a 3D view (right image). In the 3D view, the flight board is represented by a mesh.

1.3 Plan

This paper is organized as follows. First, Section 2 details the constraints of the application and presents a suitable stereo vision acquisition system. Section 3 introduces our hybrid intensity depth segmentation, a hybrid segmentation method based on both depth and intensity images. Then, Section 4 details an approach based on the Kalman filter and GNN to track multiple targets in 3D. Section 5 shows the tracking and segmentation results, which rely on an appropriate ground truth. Finally, Section 6 concludes our work and opens promising perspectives for tracking and behavioral analysis.

2 Acquisition system

In this section we will present the constraints related to our application before summarizing the suitable 3D sensors that were available on the market in 2012. Finally, we will focus on our stereo vision system and detail its configuration.

2.1 Application constraints

Several constraints had to be taken into account in the choice of the 3D camera such as the number, the size and the dynamics of the targets, the lighting conditions and the background. Each constraint is outlined below:

Number. Figure 4 shows strong bee activity in front of the beehive. The bee counter [3] showed that bee arrivals and departures can occur in significant waves. The greater the number of bees present simultaneously in the surveillance space, the more spatial occlusions (bees completely or partially overlapping others) occurred.

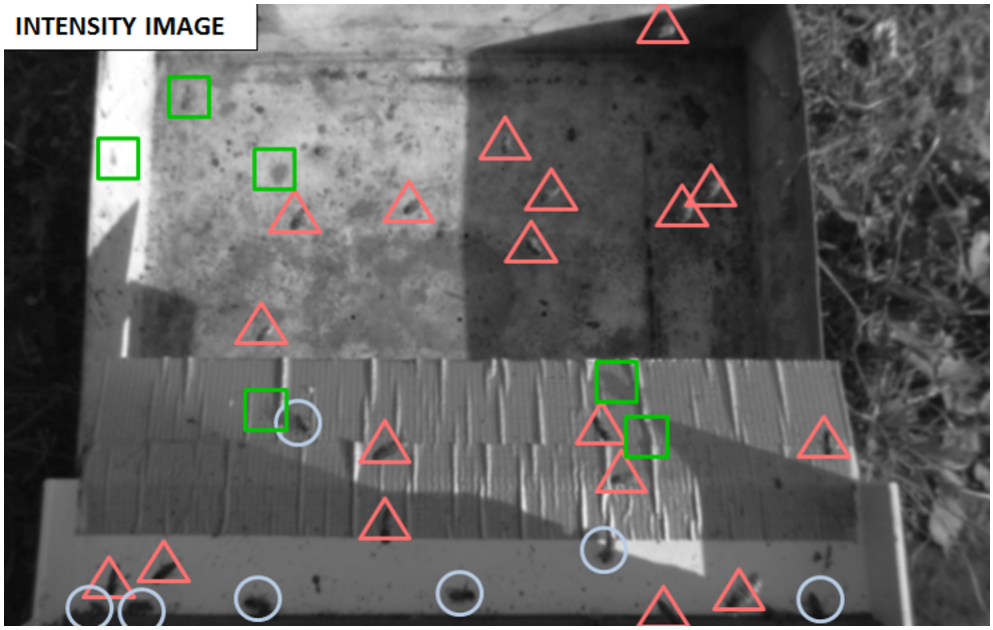

Figure 4: Intensity image of the beehive entrance. Beehive entrance captured by a stereo camera. The annotations (shapes on the intensity images) were based on manual observation of the motion (comparing sequential images) and the information provided by both the intensity and the disparity images. Here we distinguish 19 flying bees (red triangles), 7 walking bees (blue circles), and 6 bee shadows (green squares).

Size. To ensure accurate counting, the camera needs to capture the entire 50-cm-wide board from where bees enter and leave the hive. Adult bees measure on average 12 mm $\times$ 6 mm, so to detect them on the flight board, we set a limit of 8 pixels per bee on the images. Thus, Xres $= (8\,\text{pixels}/0.6\,\text{cm}) \times 50\,\text{cm} =$ 667 pixels is the minimum horizontal resolution which satisfies this small target size constraint.

Dynamic. Bee motion is highly unpredictable. They can fly at a speed of 8 m/s, so they can cross the entire flight board and only be captured on one or two images with a conventional 24-fps sensor. Even when mostly slower bees are observed around the beehive, a high-frequency capturing system is recommended. Besides, an average exposure time results in blurring due to wing movement, although this is not important in our application.

Light. The acquisitions were performed outdoors, so the lighting conditions are almost impossible to control. Images can contain more bee shadows than bees themselves. Moreover, it is worth noting that sunlight interferes with 3D sensor technologies such as infrared grid projection/sensors (e.g., Microsoft Kinect).

Background. The authors of [9] segmented bees from a white flight board, which would appear to be optimal. However in most cases, the flight board gradually becomes soiled. Our application was therefore designed to work on a textured flight board (e.g. due to dirt), which could even acquire a color similar to that of bees after a while.

Given the constraints mentioned above, we believe that in view of the high occlusion rate and the chaotic dynamics of the targets, additional data (third dimension) is required to ensure robust detection and tracking of the targets.

2.2 Candidate 3D sensors

We focused our attention on two kinds of 3D sensors (also called 2.5D sensors): time of flight (TOF) and stereo vision cameras. In contrast to homemade multiple camera systems [22], built-in 3D cameras do not require any calibration and directly provide gray (or RGB) and the corresponding depth images (also called disparity maps for stereo vision). As we will focus on stereo vision systems, additional information on TOF cameras can be found in [23].

Concerning TOF cameras, the device specifications and especially the resolutions presented in Table 2 are too low for our application since the requirement that a bee must be represented by 8 pixels is not satisfied. Nevertheless, the high frame rate of the CamBoard Nano was an interesting feature as capturing a fast moving object at a speed of 90 fps would reduce tracking failures. In this case, given the low resolution, we could focus on a smaller part of the flight board to get enough pixels to detect a bee. Finally, for stereo vision cameras, we chose the G3 EV camera that seemed to satisfy the constraints with small and fast-moving targets.

Table 2: Comparison of camera specifications

Camera	Specification	Manufacturer
Bumblebee 2	Stereo, 640×480 pixels, 48 fps	Point Grey Research (Richmond, Canada)
G3 EV	Stereo, 752×480 pixels, 50 fps	TYZX (Menlo Park, USA)
SVC	Stereo, 752×480 pixels, 30 fps	Focus Robotics (Hudson, USA)
CamBoard nano	TOF, 160×120 pixels, 90 fps	Pmdtec (Dresden, Germany)
D70	TOF, 160×120 pixels, 50 fps	Fotonic (Stockholm, Sweden)
SR4000	TOF, 176×144 pixels, 50 fps	MESA Imaging (Zurich, Switzerland)

Comparison of six camera specifications (resolution and frame rate) available from the main producers of TOF and stereo vision systems in 2012.

Figure 5 shows intensity images and depth maps acquired by the G3 EV and CamBoard Nano cameras. Image (a) is a clear RGB image while image (b) is an intensity map that is limited by the amount of light received by the sensor. The depth map (a) was well computed on highly textured areas. In reasonable conditions (targets closer than 50 cm from the camera and moving at an average speed), bee textures were efficiently captured, and consequently, a good pair matching was achieved between left and right images. The close detection range of 2 m of (b) filtered the main part of the background but not the flight board, which constitutes a significant difficulty. Moreover, the capture of depth for white or reflecting objects was not satisfactory. In addition, targets moving far from the center tended not to be clearly captured by the sensor. Finally, stereo cameras have a lower frame rate than TOF cameras due to the complexity of depth map computation. However, the G3 EV still reaches 50 fps thanks to an embedded processors unit. In conclusion, Table 3 summarizes the strengths and weaknesses of both cameras tested and evaluated relative to our application constraints. From this evaluation, the G3 EV stereo camera seems to offer the best compromise for capturing high-resolution images and depth maps at an acceptable frequency.

2.3 Stereo camera configuration

We chose a small baseline which makes the depth maps more accurate for close-range applications. Then, depending on the lenses available from the constructor, the best solution maximizing the tracking area is given by

$$\arg\max_{\alpha} a(\alpha) = \frac{(f/2)^2}{\tan(\alpha/2)} \tag{1}$$

with f being the flight board width and α the lens horizontal field of view (HFOV). An adequate de-

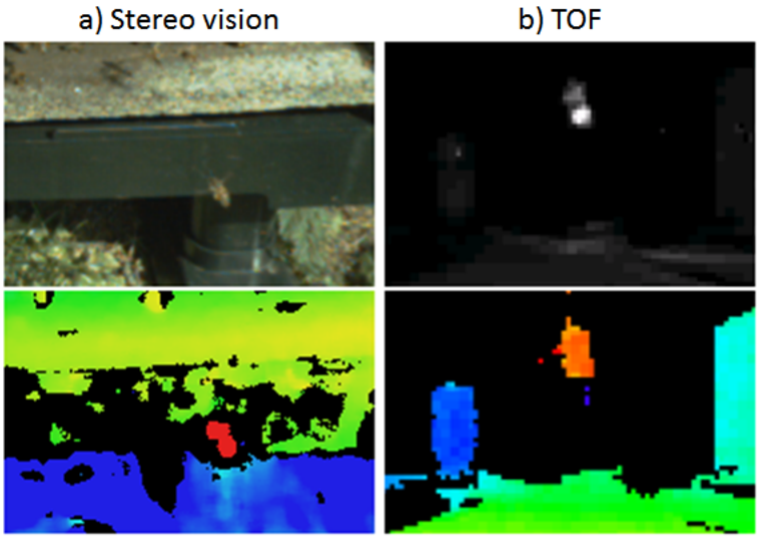

Figure 5: Images captured by two different RGB-D cameras. **(a)** Stereo vision and **(b)** TOF. Images (top row) and depth maps (bottom row) of a sample bee captured by the G3 EV (stereo vision) and CamBoard Nano (TOF) cameras.

Table 3: Comparison of capture with a TOF and a stereo vision camera

	CamBoard Nano	**G3 EV**
Frame rate	+++	++
Resolution	+	+++
Small target detection	+++	+++
Fast-moving target detection	+++	++
Depth map accuracy	++	+++
Depth map consistency	+++	++

A comparison of capture with a TOF (CamBoard Nano) and a stereo vision camera (G3 EV) depending on our application constraints.

tection of bees should be possible given the limitation of having at least 13 pixels per centimeter on the flight board. The number of pixels per centimeter on the flight board is given by

$$\text{ppc} = \frac{\text{Hres}}{2h \tan\left(\frac{\propto}{2}\right)} \tag{2}$$

with Hres as the horizontal resolution of the camera and h as the distance of the camera from the board. Finally, the chosen configuration for this application was a 3-cm baseline stereo camera equipped with 62° HFOV lenses placed at a height of 50 cm from the flight board. With this configuration, the G3 EV stereo camera acquires frames at an average frame rate of 47 fps with a variation situated between 25 and 53 fps for a given capture. Figure 6 shows the configuration of the G3 EV stereo camera targeting the flight board.

3 Hybrid intensity depth segmentation

In this section, we highlight the shortcomings of using only intensity images or disparity images to detect flying bees at the beehive entrance. Accordingly, we introduce our hybrid intensity depth segmentation

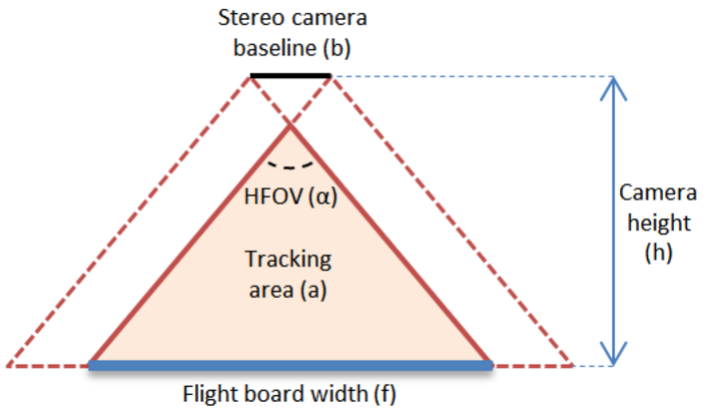

Figure 6: Stereo camera configuration.

(HIDS) method, a hybrid segmentation based on depth and intensity images.

In terms of intensity images, many motion detection methods are based on background modeling. Depending on the conditions, simple methods (e.g., approximated median filtering) can perform nearly as well as more complex techniques (e.g., Gaussian mixture models) [24]. However, most methods based on intensity images reveal their limits when used under difficult conditions. In our application, intensity values were strongly affected by recurrent and rapid changes in lighting, shadows, and reflections. When lowering thresholds for motion detection, the adaptive methods generally tend to include near-static elements in the background too quickly. Even when focusing on small temporal windows, the results are not satisfying.

Disparities were computed by a stereo pair matching algorithm. The disparity map contains holes (unmatched areas) for which there is no certainty that they correspond to a target. These holes are caused by unmatchable textures that are too uniform, too different, or simply outside the disparity range. Under satisfying conditions, flying targets are represented by a peak on the depth map, and under difficult conditions, they are represented by holes. However, holes do not necessarily indicate the presence of a target. Figure 7 shows different effects that can be observed on the depth map depending on the situation. The most common reason for holes is that the part of the background hidden by a flying target is different depending on the point from which the target is observed, so it becomes unmatchable.

The strength of our segmentation method is that it relies first on the depth map, on which potential targets (peaks and holes) are detected, and this is then confirmed using the motion calculated from the corresponding intensity images. Depending on the light, flying bees project shadows onto the flight board which may be detected as motion on the intensity map. However, it is unlikely that a hole would be observed on the depth map in an area where there is motion because a detected motion indicates a significant change in texture. Furthermore, significant changes in texture allow matching for disparity computation in most cases and do not result in holes. This therefore constitutes the strength of our method: the combination of both disparity maps and intensity images prevents false detections that are generally triggered by the shadows of flying bees.

3.1 Flying target detection

Our segmentation method is an extension of standard motion detection methods with adaptive background modeling. The main improvement is the use of the depth information to drive the adaptation of the background intensity model.

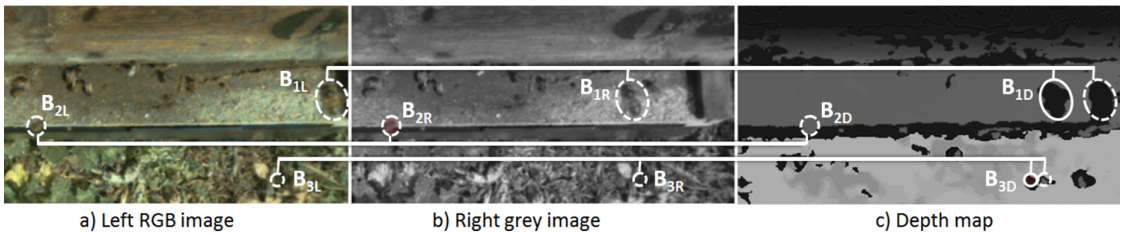

Figure 7: Different depth map effects. Different depth map effects observed with the G3 EV camera: **(a)** left RGB image, **(b)** right grey image, **(c)** depth map. The B_1 bee (high-speed target) is not matched between (a) and (b), and this results in the shadow effect to the right on (c). Moreover, its high elevation from the background produces an additional shadow effect on the left. The B_2 bee (non-moving target) is correctly matched, but being close to the background, no shadow effect is observed. The B_3 bee (normal-speed target) is half matched and produces a double-shadow effect for the same reasons as B_1.

The stereo camera provides a pair of grayscale images (left and right) and a corresponding disparity map. Below, $I_{t,u,v}$ refers to the intensity of the pixel at time t and position (u, v), while $D_{t,u,v}$ refers to the distance from the camera at time t and position (u, v). The objective here is to compute two binarized masks based on I and D: a determined depth target mask $DDTM$ and an undetermined depth target mask $UDTM$. The DDTM represents targets with depth information that may be recovered, and UDTM represents targets with no direct recoverable depth information.

3.1.1 DDTM

The determined depth target mask DDTM is based on background subtraction between D and the computed depth background image, DBG, (see below):

$$\mathrm{DDTM}_{t,u,v} = \begin{cases} 1 & \text{if } D_{t,u,v} - \mathrm{DBG}_{u,v} > \Delta d \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where Δd is a threshold. A morphological opening is then applied to remove the noise in the depth map D.

The depth background image DBG is computed over several frames by a non-evolutive temporal median:

$$\mathrm{DBG}_{u,v} = \mathrm{median}\{D_{t_0,u,v}, D_{t_0+\Delta t,u,v}, ..., D_{t_0+k\Delta t,u,v}\}. \tag{4}$$

Unlike intensities, disparity values are generally stable over time regardless of changes in lighting. A small jitter effect (few millimeters) is caused by imperfections in intensity image matching, but the values remain around an average value that corresponds to the real depth. The quality of the depth background DBG depends on the crowding condition. An increase in the frame number k used in the median computation and the increase in the time Δt between two frames improve the robustness of the depth background computation with respect to passing (flying or walking) targets.

3.1.2 UDTM

In order to compute the UDTM, we first computed an undetermined depth mask UDM, which contained regions of the depth map D with undetermined depth and excluded regions of the depth background

DBG with undetermined depth. UDM was computed as the intersection of D and DBG:

$$\mathrm{UDM}_{t,u,v} = \begin{cases} 1 & \text{if } (D_{t,u,v} = e) \cap (\mathrm{DBG}_{\mathrm{u,v}} = e) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where e is the value assigned by the stereo camera to pixels that have an undetermined depth and $\cap$ is the logical conjunction operator.

Then, an intensity absolute motion mask $IAMM$ was computed based on the absolute difference between an intensity background image IBG (see below) and I. A morphological closing was applied to enlarge potential motion regions and merge them with their close neighbors. IAMM was computed as

$$\mathrm{IAMM}_{t,u,v} = \begin{cases} 1 & \text{if } ((\,|\,\mathrm{IBG}_{\mathrm{t,u,v}} - I_{\mathrm{t,u,v}}\,|\,\oplus S_1 \ominus S_2) * M) > \Delta m, \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\oplus$ is a dilation using the structuring element S_1, $\ominus$ is an erosion using the structuring element S_2, $*$ is a convolution with the mean filter M, and Δm is the threshold for binarization. We chose S_1 to be bigger than S_2.

Finally, UDTM was computed as the intersection of UDM and IAMM:

$$\mathrm{UDTM}_{t,u,v} = \mathrm{UDM}_{t,u,v} \cap \mathrm{IAMM}_{t,u,v}. \tag{7}$$

The evolutive temporal median intensity background IBG used for the computation of IAMM was initialized as follows:

$$\mathrm{IBG}_{t_0,u,v} = \mathrm{median}\{I_{t_0,u,v}, I_{t_0+\Delta t,u,v}, ..., I_{t_0+k\Delta t,u,v}\}. \tag{8}$$

The intensity relative motion mask $IRMM$ corresponds to the relative motion map. It was computed by

$$\mathrm{IRMM}_{t,u,v} = \begin{cases} 1 & \text{if } (\,|\,I_{\mathrm{t,u,v}} - I_{\mathrm{t\text{-}1,u,v}}\,|\,\circ S_3) > \Delta\mathrm{rm} \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

where $\circ$ is a morphological opening using the structuring element S_3 to enlarge potential motion regions, and Δrm is a threshold for binarization.

The foreground mask FG included all the potential targets except areas that exhibited no motion. FG was computed as

$$\mathrm{FG}_{t,u,v} = \mathrm{EDDTM}_{t,u,v} \cup (\mathrm{UDTM}_{t,u,v} \cap \neg\,(\mathrm{UDTM}_{t,u,v} \cap \mathrm{IRMM}_{t,u,v})) \tag{10}$$

where $EDDTM$ was obtained by applying a morphological dilation to DDTM to be sure that the whole targets were included in FG.

$$\mathrm{EDDTM}_{t,u,v} = \mathrm{DDTM}_{t,u,v} \oplus S_4 \tag{11}$$

FG was used to perform a selective adaptation of IBG as defined by

$$\mathrm{IBG}_{\mathrm{t+1,u,v}} = (\delta \cdot \mathrm{FG}_{\mathrm{t,u,v}}) \cdot I_{\mathrm{t,u,v}} + ((1-\delta\,) \cdot \mathrm{FG}_{\mathrm{t,u,v}}) \cdot \mathrm{IBG}_{\mathrm{t,u,v}} \tag{12}$$

where δ is the learning rate used for the adaptation.

Figure 8 illustrates the segmentation process described above. Our segmentation works especially well in outdoor conditions because typically non-uniform textures are present, which is favorable for the stereo matching algorithm.

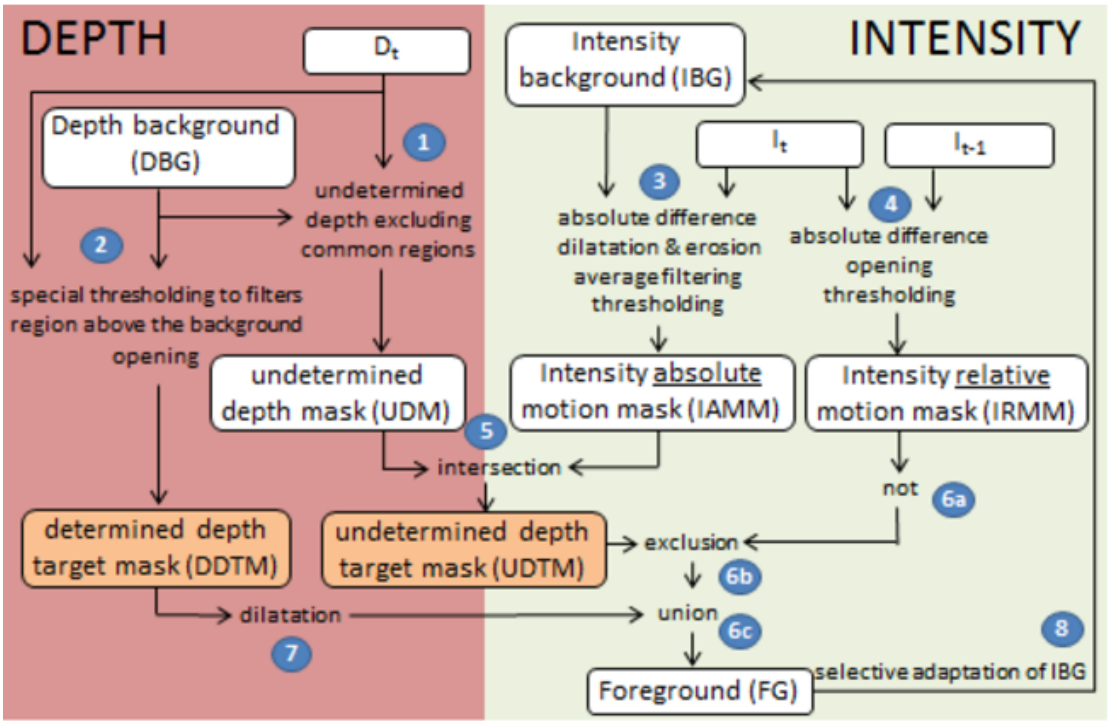

Figure 8: Hybrid segmentation process. A hybrid segmentation process based on both intensity and disparity images. The numbers correspond to the order of the steps.

3.2 Target extraction

The previous step provides two binarized masks, DDTM and UDTM, which represent targets with recoverable depth information and targets with no direct recoverable depth information, respectively. The centroid (center of mass) of each region of DDTM and UDTM is associated with a target. Using DDTM, the depth can be recovered by computing the median value of the depth values corresponding to the region in the depth map. Moreover, an ellipse is approximated to each region, giving information on the orientation and the size of the target. Figure 9 shows examples of extracted ellipses for the following cases: available and unavailable depth for targets over a clear and a cluttered background.

Raw statistics on depths and the size of ellipses was collected in a preliminary segmentation without constraints. We identified an exploitable relation between the lengths of the major axis of the ellipses and the depths. This relation appeared to be almost linear at this depth scale, although it cannot be linear on a bigger scale. Figure 10 shows that, using a polynomial regression, this relation can be approximated by two functions (mean value μ_d and standard deviation σ_d). Thus, we considered that observations from our segmentation fitted this model. The increase in the distance between an observation and the model may correspond to a false alarm. The degree of truth that an observation belongs to false alarms is given by the following membership function:

$$m_d^{FA}(s) = 1 - e^{-\frac{(s-\mu_d)^2}{2\sigma_d^2}} \tag{13}$$

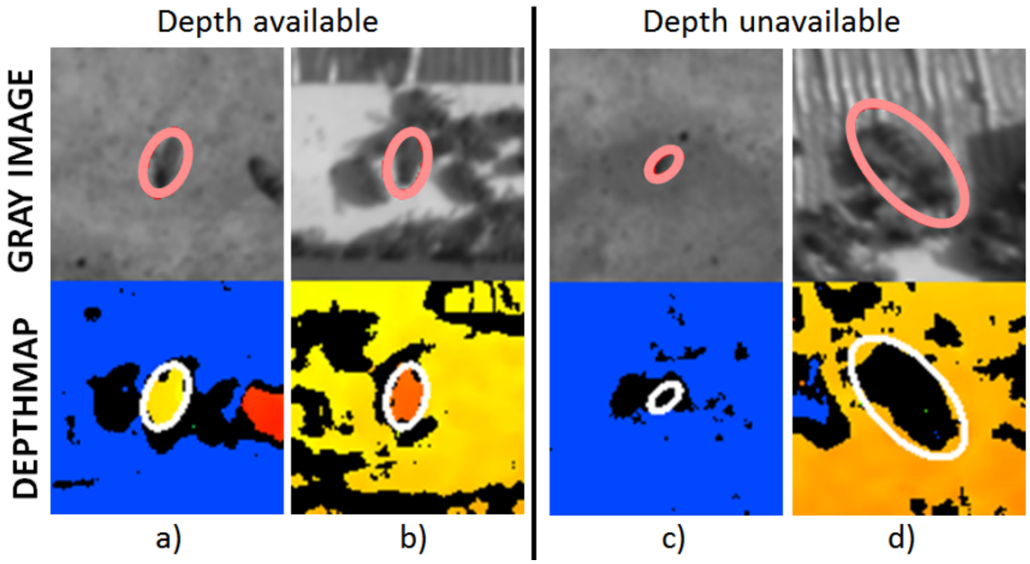

Figure 9: Four segmented targets. Four segmented targets and corresponding extracted ellipses for four pairs of intensity and disparity images. Depth is available for pairs **(a)** and **(b)**, but not for **(c)** and **(d)**. Depth is unavailable for (c) and (d) as the target in (c) is far away and too small, and the target in (d) is too close (out of the disparity range). Contrary to (a) and (c), (b) and (d) are in cluttered conditions (motion in the background).

where d is the depth and s is the size of the target. Concerning targets without depth information, our initial idea was to approximate their depth from their size, but we invalidated this idea given the standard deviation of the models, which was generally quite high.

4 Multi-target tracking in 3D

In this section, we propose an approach based on the Kalman filter [25] and Global Nearest Neighbors [26] to achieve multi-target tracking in 3D (called 3D-GNN below). Each target was associated to a Kalman filter, which was used to estimate the trajectory based on incoming observations. Then, GNN associated uncertain measurements with known tracks.

The trajectory of a target can be estimated by different methods, including the extended Kalman filter [27] and particle filters [20]. Both are particularly suitable for non-linear systems, and particle filters belong to the track-before-detect approach. The particle filter (PF) approach was not used in our work for two reasons: first, it is difficult to obtain a reasonable computation time for multiple targets (up to 18 in our application) and our prototypes had difficulty maintaining tracks for multiple close targets. Secondly, our segmentation is efficient enough to adopt a detect-before-track approach. Despite the apparent rough dynamics of bees, we acquired frames at a sufficiently high frequency (about 47 fps) to allow us to assume a constant speed model. Thus, an approach based on the standard Kalman filter was suitable for our application.

In terms of data association methods, there are full Bayesian approaches such as the Joint Probabilistic Data Association (JPDA) filter [28] or the MHT [26]. GNN is a non-Bayesian approach which consists in computing a maximum likelihood estimate from a possible set of data association solutions. In view of our challenging dataset (Section 2.1 lists the constraints of our application), we considered that our HIDS method was reliable enough to provide clear 3D positions for targets among the clutter, which

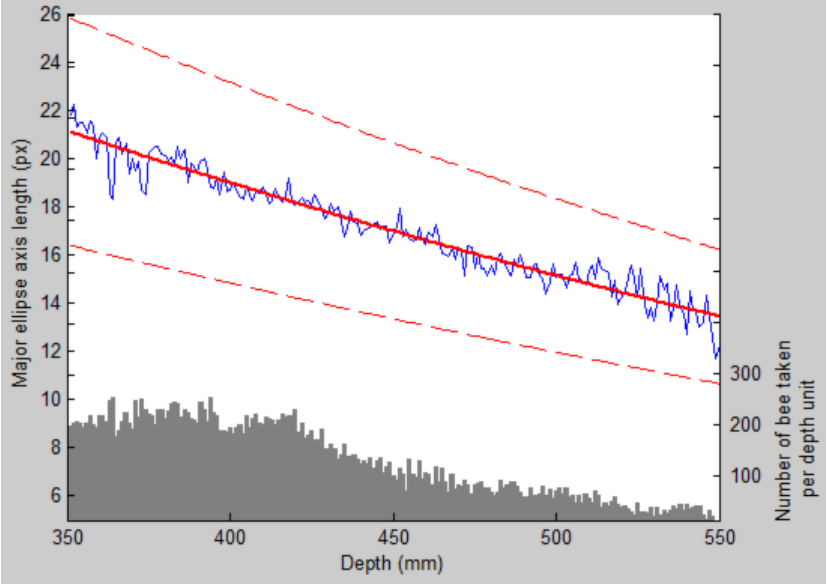

Figure 10: Size/depth relation of a target. The relation for over 50,000 segmented targets between the major ellipse axis (size of the target) and the depth. The histogram shows the number of bees per depth unit. The thin plain curves represent the length of the median axis for the given target sample. The thick and the dashed curves correspond to the estimated models of the relation between axis length and depth, and depth standard deviation, respectively.

corresponded to a high signal-to-noise ratio. Table 4 summarizes common tracker complexities according to the signal-to-noise ratio. Considering the need for real-time measurements in our application, we focused our study on GNN. In addition, we also employed MHT to perform comparisons.

Table 4: Tracker complexities

	Low SNR	**Medium SNR**	**High SNR**
Low computation	GNN	GNN	GNN
Medium computation	MHT	GNN or JPDA	GNN
High computation	FP or MHT	MHT	Any

Tracker complexities in relation to the signal-to-noise ratio (SNR) condition.

4.1 Kalman filter model

To ensure coherent tracking in 3D, the model (state and measure vectors) was defined in camera coordinate space (3D Euclidian space) where the reference sensor (the left imager of our stereo camera) was located at position (0,0,0). The projection of observations from the image coordinate space onto the camera coordinate space is explained in Section 4.2.

Let $Y_{1:n}$ be a series of observations corresponding to a target from time 1 to n. For a given step k, an observation is defined by the vector $Y_k = [x, y, z]^T$, and the estimated state of a target is defined by the vector $X_k = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$ combining its 3D position and velocity. Figure 11 lays down the recursive mechanism of the KF for the estimation of the state vector X_k.

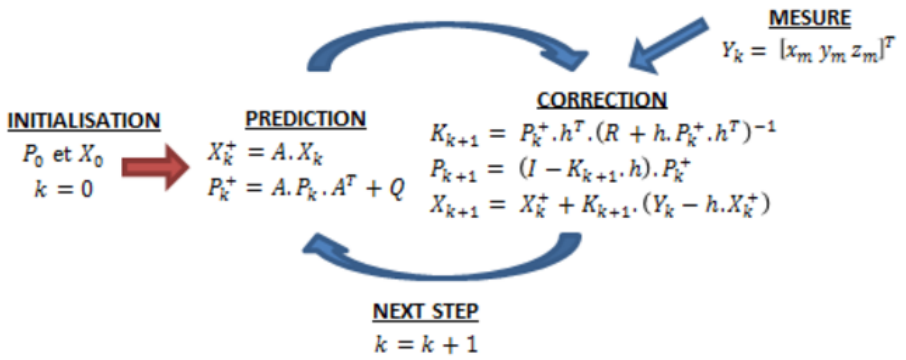

Figure 11: Recursive mechanism of the KF. Recursive mechanism of the Kalman filter for the estimation of X.

4.2 Projection onto a 3D Euclidian space

The segmentation step provides all of the targets with a position (u, v, d), which is expressed in the image coordinate system, where u and v are the column and the row indexes, respectively, and d is the computed depth for that point. The measurement model for our KF works in the 3D Euclidian camera coordinate system, the projection of (u, v, d) from the image onto the camera coordinates (x, y, z), is given by:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = d \begin{bmatrix} \frac{u - cu}{f_u} \\ \frac{v - cv}{fv} \\ 1 \end{bmatrix} \tag{14}$$

with (cu, cy) as the stereo camera calibration parameters, which refer to the pixel coordinates of the principal point, and (f_u, f_v) as the focal lengths in pixels along the x- and y-axis, respectively.

4.3 Missing depth management

Given the segmentation method used and the reliance placed on the ability to recover the corresponding depth, some targets were clearly detectable in 2D (u, v), but the depth could not be recovered or considered reliable (see Section 3). An observation cannot be projected onto camera coordinate space without a depth d. Moreover, as mentioned in Section 3.2, the relation between the size of the segmented ellipse and the depth is not reliable enough to directly infer the depth from the size. To this end, we propose the following missing depth management method based on estimation and reprojection.

When the depth was missing, the estimated z from the KF was temporally used as d for the projection in (14). Consequently, a complete observation could be provided for the KF in the update step. However, in order to avoid degeneration, the number of estimated projections was limited. When the depth information became available again for a later observation, a new estimation of the depth was interpolated (using cubic spline interpolation [29]) over the window during which the depth was missing. Then, because of the change of z, all the observations associated with the KF over that window were reprojected using the new z value. Finally, the trajectory was re-estimated using the new reprojected observations in the KF, starting from the state where d was unavailable. The advantage of this method is that it keeps estimations as close as possible to the available data. Algorithm 1 illustrates the process.

Algorithm 1: Tracking with reprojection mechanism

while *Observation available for a target* **do**
    1) Prediction
    2) Observation
    if z *missing* **then**
        % Enter in a temporary estimation loop
        while *Z missing* **do**
            a) Projection (using the last predicted z)
            b) Update
            c) Prediction
            d) Observation
        end while
        Interpolate z from the last and new observed z
        Reproject observations with new interpolated z
        Reapply the KF with new reprojected observations
    end if
    3) Projection
    4) Update
end while

As mentioned before, a track can be initialized with an unknown depth. In this case, an arbitrary depth is given to the new KF. Then, when an observation with a known depth is finally associated, all the previous observations associated with the KF are reprojected using that known depth.

4.4 Multi-target assignment

Our approach is based on GNN, which is detailed in this section. GNN is a widely used method for data association and has the advantage of having a relatively low degree of complexity (see Table 4). We also address in this section the MHT, which is used in Section 5 as a basis for comparison with GNN.

4.4.1 Global Nearest Neighbor

GNN is used to handle track instantiations, destructions, and associations with observations. The assignment matrix $A[c_{i,j}]$ represents all the possible associations and the costs generated by these associations. A includes the possibility for each observation to be associated with an existing track, not to be associated with any track, or to be associated with a new track. $c_{i,j}$ is the cost for the observation i to be assigned to the possibility j. The best configuration of associations is the solution that minimizes the total cost, which corresponds to to the given linear assignment problem, which can be solved using, for example, the Hungarian method:

$$\begin{array}{ll} \sum c_{ij}^2.x_{ij} & x_{ij} \in \{0\ 1\} \\ \sum_i x_{ij} = 1 & \sum_j x_{ij} = 1 \end{array} \tag{15}$$

The Mahalanobis distance d^2 between an observation and track is given by

$$d^2 = (Y - MX^+)' S^{-1} (Y - MX^+) \tag{16}$$

$$S = ME^+M' + Em \tag{17}$$

where $(Y - MX^+)$ is the innovation, with Y as the measure, M as the measurement matrix, and X^+ as the *a priori* predicted position. S is the prior covariance of innovation with Em as the measure noise

matrix and E^+ as the predicted noise covariance matrix. Then, the association cost $C^A_{i,j}$ between an observation and an existing track is given by

$$C^A_{i,j} = C_K + C_S \tag{18}$$

$$C_K = \ln(\sqrt{|S)|}) - \frac{N\ln(2\pi) + d^2}{2} \tag{19}$$

$$C_S = \ln(\frac{P_D}{1 - P_D}) \tag{20}$$

where P_D is the probability of the observation to be a true target and N is the dimension of the state vector (here $N = 6$). $P_D = 1 - P_{\text{FA}}$ if the depth of a target is recoverable. P_{FA} is given by the membership function m^{FA} (13) defined in Section 3. If the depth is not recoverable, a neutral value (e.g., 0.5) is affected to P_D. The respective costs $C^{\text{FA}}_{i,j}$ and $C^{\text{NT}}_{i,j}$ when an observation is found to be a false alarm or a new track are given by

$$C^{\text{FA}}_{i,j} = -\ln(dC) \tag{21}$$

$$C^{\text{NT}}_{i,j} = -\ln(dN) \tag{22}$$

where dC and dN are respectively the false alarm and track apparition density functions relative to the surveillance space. As an example, dN can be modeled by a map where each position is weighted by its distance from the closest potential target apparition point.

Finally, the associated observations are processed using the associated Kalman filter, and non-associated observations become candidates to initiate a new track. A track is destroyed if it is not associated to an observation three times in succession or if the sum of its historical association costs reaches zero. More details on GNN are given in [26].

4.4.2 Multiple Hypothesis Tracking

In contrast to GNN, which only maintains the single most likely hypothesis, MHT differs the tracking in order to consider alternative hypotheses within a limited period of time. The hypotheses are stored in a decision tree, which grows by one level at each step of the tracking. In order to avoid a combinatory explosion, a pruning process is applied to keep the tree at a reasonable size. In addition, only a limited number of best hypotheses are considered at each step (see the Murty algorithm). The score of a hypothesis corresponds to the sum of all the associated costs of tracks belonging to that hypothesis from the creation of the track. Concerning the association costs, MHT shares a common base with GNN (see Section 4.4.1). Finally, a fusion mechanism checks and deletes similar tracks, which are likely to follow the same target. More details on MHT and its implementation are given in [30].

5 Results

This section deals with the evaluation of our HIBS segmentation using a dedicated segmentation ground truth and, in addition, the evaluation of our 3D-GNN tracking using a second dedicated ground truth. The establishment of these two ground truths for segmentation and tracking was essential since they correspond to different constraints and needs, which are detailed below.

5.1 Segmentation evaluation

5.1.1 Segmentation ground truth

The evaluation of our segmentation relies on two indicators: true negatives (miss detections) and false positives (false alarms). Fully annotated images are required in order to compute these indicators, which means that every flying bee visible on the images has to be marked manually. This process is time-consuming and is not necessarily reliable. The sequences used for the evaluation were captured in natural conditions, and the images often presented ambiguous situations from a human point of view. To ensure the quality of our segmentation ground truth, we adopted a triple-blind annotating process. Each of the frames was annotated by three different people. We then chose the following approach to interpret the multiple annotated data: a target was considered to be real if at least two people marked it with an ellipse, and an ellipse was considered to belong to a common observation when the center of one ellipse was included in another. Based on this method, 4.8% of the marked ellipses (45 out of 928 ellipses) were considered to be mistakes due to human error. Although it requires three times more work, we chose to focus on quality in spite of the quantity. Our segmentation ground truth consists of 500 frames randomly extracted from nine 15-min sequences captured under different conditions (see Figure 12). The random selection of frames from the nine sequences ensured a complete representation of the conditions used with our application.

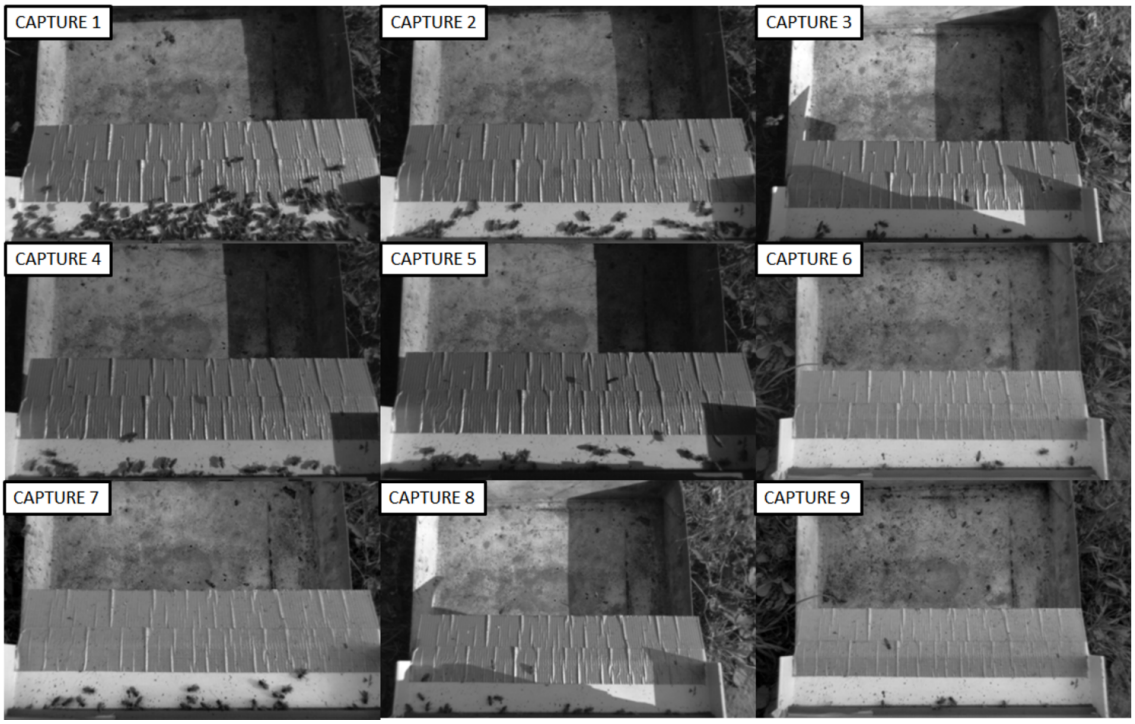

Figure 12: Ground truth images. Images extracted from the nine captures used for the ground truth (GT). These captures were performed in order to cover a large variety of conditions (activity, nervosity of the bees, light, reflections, and shadows).

5.1.2 Segmentation results

Table 5 presents our segmentation results and details the acquisition conditions for each capture. First, it shows that unstable light and intense shadows tended to increase the number of false alarms. Secondly, a high activity may increase the number of misdetected targets. Table 6 compares our hybrid segmentation

with a segmentation that only uses depth information. All the results shown were obtained from a comparison between the segmented targets and those annotated in the ground truth. The importance of using a hybrid segmentation based on intensity and motion is clearly highlighted in Table 6. The depth-only approach gave rise to just a few false alarms because it detects only stereo pair-matched targets. Nevertheless, it missed all of the targets that were beyond the matching range, which was nearly 30% in our case. Our hybrid segmentation recovered more than 95% of the bees annotated in the ground truth, although it is still hampered by quite a high false alarm rate (19%). Only 10% of the false alarms had a recoverable depth. False alarms were mostly located in crowded areas, as shown in Figure 13; here, the flight board was mainly responsible for the false alarms. Constant motion was generated by bee shadows and walking bees. This motion, which is identified on the intensity image, could potentially lead to a wrong decision being made during the segmentation process when it corresponds to unknown depth areas.

Table 5: Detailed segmentation results

Capture	Activity	Light	Shadows	Camera	TN rate(%)	FP rate(%)
Capture 2	++	VU	+++	L	11.02	28.81
Capture 5	+	U	++++	L	0.00	27.97
Capture 7	+	U	+++	L	0.00	17.80
Capture 8	++	S	+++	H	6.78	16.10
Capture 1	++	S	+++	L	1.69	15.25
Capture 4	++	S	+++	L	0.00	9.32
Capture 3	+++	S	++	H	8.47	5.93
Capture 6	++	VS	+	H	0.85	3.39
Capture 9	++	VS	+	H	0.00	0.00

Segmentation results per capture and acquisition condition. Activity (average number of flying bees per frame): +, less than 1; ++, between 1 and 8; +++, more than 8. Light: VU, very unstable; U, unstable; S, stable; VS, very stable. Shadows, intensity of shadows. Camera: L, low position; H, 10 cm higher. TN rate, true negative rate; FP rate, false positive rate.

Table 6: Overall segmentation results

Method	True negative rate (%)	False positive rate (%)
HIDS	4.15	19.54
Depth only	48.09	3.78

Average segmentation results for the nine captures.

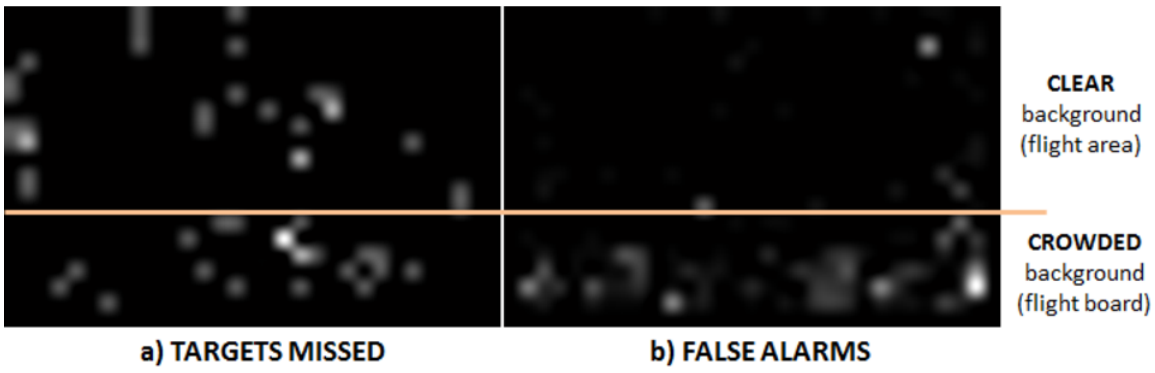

Figure 13: Spatial distribution of TN and FP. Normalized maps representing the spatial distribution of **(a)** undetected targets and **(b)** false alarms. False alarms were uniformly spread over the space, in contrast to false alarms, which were mostly located on the flight board.

5.2 Tracking evaluation

5.2.1 Tracking ground truth

The evaluation of our tracking relies on one indicator: track coverage. An exhaustive ground truth of bee trajectories at the beehive entrance would be extremely difficult to build at the scale of our application. We therefore chose to create a semi-simulated ground truth for our tracking evaluation. First, we manually annotated a base of 80 trajectories in 3D (corresponding to 3,640 positions). These annotated tracks covered more than 1,000 frames. Despite this large number, annotation was feasible because the frames were only partially annotated (focusing on one or few tracks only), unlike the segmentation ground truth that would have required fully annotated frames. We then recreated virtual scenarios by replicating and time shifting tracks randomly picked from the base of real trajectories. This method offers the advantage of building realistic scenarios while controlling basic parameters such as the number of targets. False alarms and miss detections were also added to the scenarios based on the amount and the distributions identified by the evaluation of the segmentation (Section 5.1.2). A large majority of the false alarms had no recoverable depth, which constituted the major difficulty of our taking in 3D. The probability of recovering depth on a false alarm was set to 0.1 (given by the segmentation evaluation). Naturally, occlusions due to targets crossing paths were also taken into account by removing targets that were known to be hidden from the point of view of the camera. Figure 14 shows the tracking ground truth and examples of the scenarios generated. Finally, for each observation, we defined the probability of detection as $P_D = 1 - m_d^{\text{FA}}(S \sim \mathcal{N}(\mu_d, \sigma_d))$, where m^{FA} is the membership function defined by (13), μ_d and σ_d are the mean and standard deviation functions of d, and d is the depth of the target. To summarize, the tracking ground truth contained randomly rearranged real 3D tracks of flying bees moving within the surveillance space. The surveillance space was a 3D pyramid (40×40 cm base, height of 40 cm) located at the beehive entrance.

5.2.2 Tracking results

This section compares the tracking results for GNN and MHT under different conditions (2D and 3D). GNN is a particular case of MHT that keeps only the best hypothesis for each step. In this section, P defines the depth of the MHT tree (pruning level), and H is the number of best hypotheses considered at each node of the tree. A track was considered to be well recovered when the associated observations matched at least 90% of the original track (the margin of 10% corresponds to the potential delay for track initialization and destruction). The scenarios were generated with an element of randomness. To ensure our results were relevant, we ran all of the following experiments on 100 distinct scenarios, which provided an acceptable stability for the averages.

The details of the basic configuration (when not redefined) used for our evaluation are given below. Miss detection and false alarm rates were set at 11.0% and 28.8%, respectively, which correspond to the most difficult sequence found in the segmentation evaluation (capture 2). A similar distribution was seen with dC, as illustrated in Figure 13. Concerning dN, as the targets do not only appear along the edge of the surveillance volume but also at the beehive entrance and from below the beehive, a uniform distribution was considered. In order to highlight our contributions, we ran the following four experiments.

Experiment 1: A comparison in 3D-GNN with more complex trackers (3D-MHTs). As Figure 15 shows, the computational complexity of MHT increased with the number of targets. Considering the amount of data that had to be processed and the need for real-time acquisition in our application, we preferred not to use MHT. Figure 16 shows that, in our application, MHT did not perform much better than GNN, especially with a large number of targets, which can be explained by the relatively low number of false alarms and miss detections.

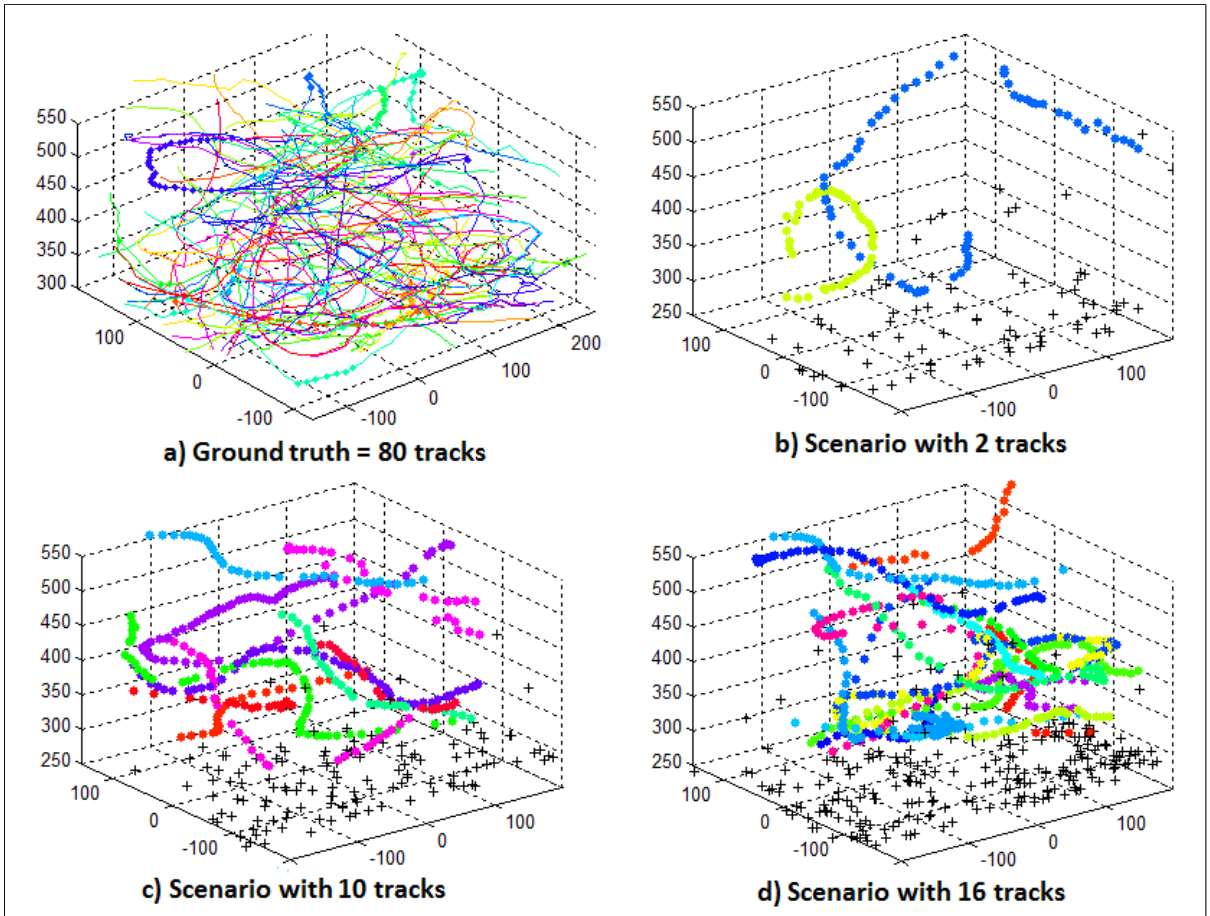

Figure 14: Tracking ground truth. Tracking ground truth and generated scenarios. The 80 manually labeled 3D tracks are shown in **(a)** where the thick dots correspond to observations with no recoverable depth. Examples of generated scenarios with 2, 10, and 16 targets are shown in **(b)**, **(c)**, and **(d)**; black crosses correspond to false alarms, which were similarly distributed, as illustrated in Figure 13. Ninety percent of false alarms are defined with no depth and are displayed at $z = 250$.

Experiment 2: A comparison of trackers taking advantage of the third dimension (3D-GNN and 3D-MHTs) with trackers relying only on 2D data (2D-GNN and 2D-MHTs). In this experiment, 2D data were obtained by projecting the 3D data onto a 2D plane. Figure 16 shows the dominance in 2D of MHTs (2D-MHTs) over GNN (2D-GNN), especially with a restricted number of targets. However, it also confirms the importance of the third dimension; 3D-GNN provided better results than complex 2D-MHT trackers. Therefore, in the following experiments, we focused only on 3D-GNN.

Experiment 3: A comparison of 3D-GNN with and without the use of the reprojection method. Contrary to a basic segmentation, our hybrid intensity/depth segmentation also recovered targets with no recoverable depth, which then became candidates for reprojection. In this experiment, an attempt was made to remove all of the targets that were not fully defined in 3D, with the effect that the occurrence of false alarms was reduced, but this also removed some correct detections. The results shown in Figure 17 confirm the need to use targets with no recoverable depth to achieve a complete tracking. It is clear that in 3D the difficulties are related to observations with no depth; however, these observations are needed in order to consider the corresponding tracks. When using only full 3D defined observations, the increased number of targets had almost no effect on the performance. Therefore, in our application, the tracking results were directly driven by the quality of the segmentation.

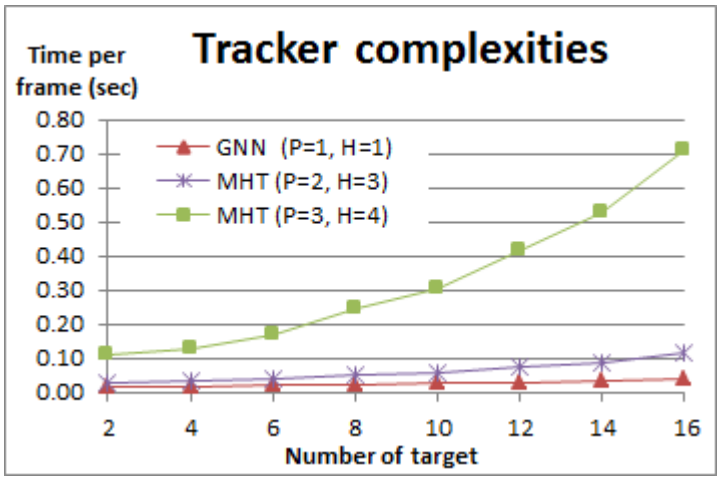

Figure 15: Tracker complexities. Comparison of computation times for GNN and MHT. Experiments were run on CORE i5 2.4 GHz.

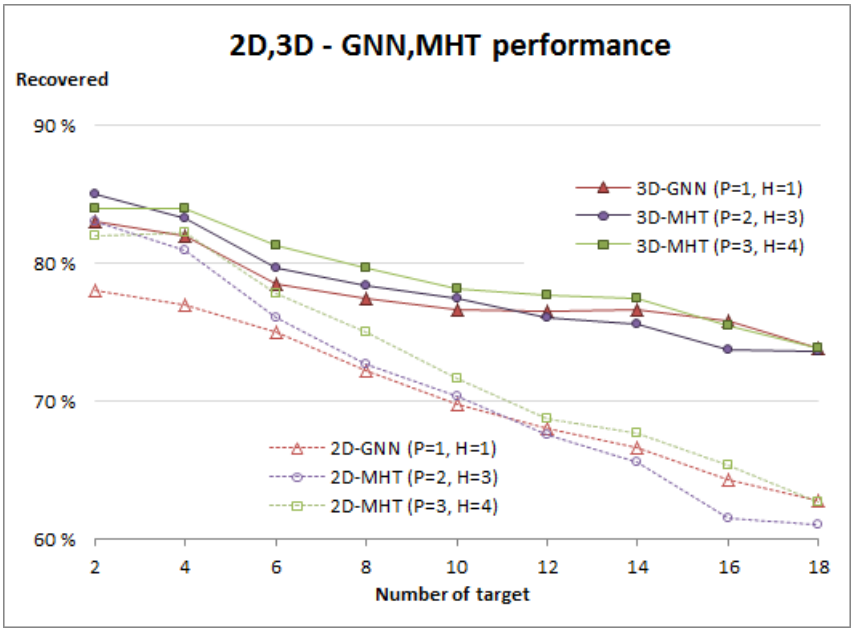

Figure 16: 2D and 3D performance of GNN and MHT. Comparison between the 2D and 3D approaches of GNN and MHT, where H is the number of hypotheses, and P is the pruning level. For each number of targets (2 to 18), the results were computed on an average of 100 different scenarios, i.e., a total of 900 different scenarios.

Experiment 4: Tests of the robustness of 3D-GNN with different probabilities of miss detection (MD) and false alarm (FA). In addition to the normal conditions, we tested our 3D-GNN when the number of false alarms and the number of miss detections were doubled. Figure 18 shows that our tracker had more problems with miss detections than false alarms. This confirms our segmentation strategy, which offers an optimal MD/FA ratio.

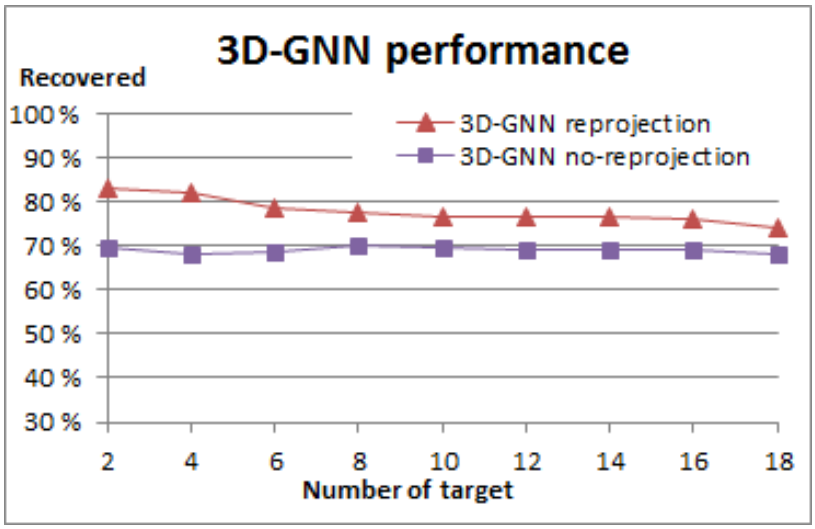

Figure 17: 3D-GNN reprojection performance. Comparison of 3D-GNN with and without the use of our reprojection method.

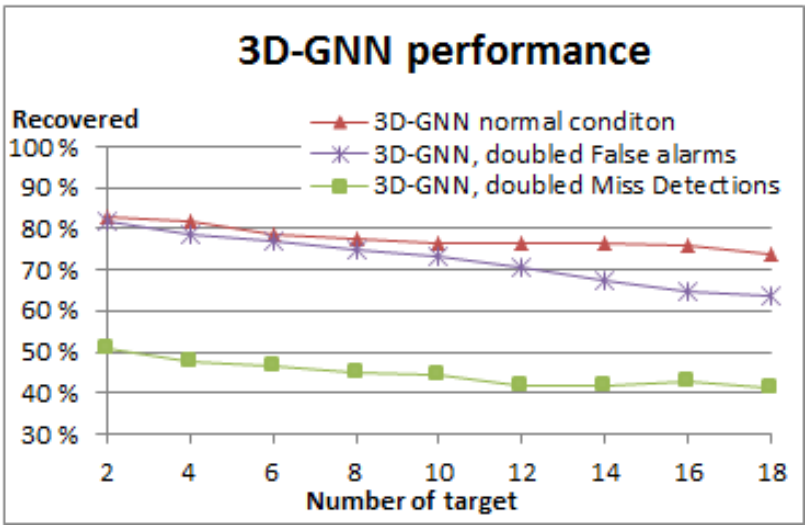

Figure 18: 3D-GNN performances under different conditions. Comparison of 3D-GNN under different conditions of segmentation.

6 Conclusions

In this paper, we have presented a system designed to acquire and track small, fast-moving targets in 3D. Our application for tracking honeybees in natural conditions is subject to many constraints (real-time acquisition, number, size and dynamics of the targets, lighting, and gradual soiling). Accordingly, after a comparison of potential suitable 3D acquisition systems (time of flight and stereo vision), we chose the G3 EV stereo vision camera. The advantage of the G3 EV is that it combines a high resolution of 752×480 pixels with a sufficiently high frame rate of 47 fps, which is sufficient to track bees.

Moreover, we propose a complete detect-before-track chain to track the targets in 3D space. To this

end, we developed a hybrid 3D segmentation method called hybrid intensity depth segmentation. Our HIBS relies on both depth and intensity images and therefore works in completely natural conditions. It outperforms the state-of-the-art methods, which mostly use intensity images only. Furthermore, our HIBS segmentation has the advantage of recovering targets with no recoverable depth, which is essential for maintaining the corresponding tracks. Our segmentation was evaluated by relying on a robust ground truth. The triple annotation process revealed that 4.8% of bees were incorrectly marked due to human error. The evaluation of our segmentation results with respect to the ground truth resulted in 4.15% of miss detections and 19.54% of false alarms. The false alarms were mainly located in complex areas such as a crowded flight board.

Each target was associated a the Kalman filter, which was used to estimate the trajectory based on incoming observations. A data association method, either Global Nearest Neighbor or Multiple Hypothesis Tracking, was employed to associate uncertain measurements to known tracks. In addition, a mechanism of temporary reprojection was used with observations for which depth information was missing. We based our tracking evaluation on a semi-simulated ground truth that relied on annotated trajectories in 3D. As expected with any tracker, the efficiency decreased as the number of targets increased. Among our captured sequences, we identified some situations with up to 18 targets. Even in these conditions, thanks to our robust HIBS segmentation, GNN provided relatively good results with respect to MHT and considering its computational complexity. In addition, the use of the third dimension, which is the strong point of our application, largely compensated for the choice of GNN, which in fine remains a simple but fast tracker. Moreover, we have shown that when reprojection is not taken into account in the tracking process, the results are much less satisfying.

In relation to the short-term perspectives, the assignment process is a constraint by gating. Currently, gating is independent from target location. However, the distance between a target and its predicted position is not uniformly distributed over 3D space. Bees arriving at the beehive entrance are less prone to sudden changes of direction or velocity. It would be interesting to add an adapted gating process depending on the location of the target. A stereo camera provides a partial topology of the scene, so the 3D position of interesting elements (flight board, entrance) could be recovered.

Concerning the longer-term perspectives, biologists are interested in high-level applications such as abnormal behavior detection. Such applications include many parameters and require robust models from which observations can then be compared. In this context, the environmental platform presented in Section 1 offers encouraging perspectives. Thanks to the modules under development (e.g., air quality monitors) and existing modules (e.g., counter, weather monitor), information of a different nature could be compared and used to model behavior. On the one hand, low-level behavior models could focus on individual bee trajectories, for example, a tracker that takes into consideration environmental parameters to adapt motion models for estimation. On the other hand, more general models could focus on colony activity such as abnormal colony behavior based on some simple rules (e.g., low activity during a sunny day). The authors of [31] demonstrated that the composition of agricultural landscapes influences life history traits of honeybee workers. It would be interesting to find a correlation between their observations and individual or general behavior detected in the trajectories of bees.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This work was supported by the European Regional Development Fund (contract: 35053) and the Poitou-Charente region. The videos used in this work were taken during autumn 2012 on the INRA site (National Institute for Agronomy Research) at Magneraud. We would like to thank INRA's biologists for their support and expertise in helping to collect exploitable data under different conditions (e.g., activity, weather).

References

1. C Vidau, M Diogon, J Aufauvre, R Fontbonne, B Viguès, J Brunet, C Texier, D Biron, N Blot, H El Alaoui, LP Belzunces, F Delbac, Exposure to sublethal doses of fipronil and thiacloprid highly increases mortality of honeybees previously infected by *Nosema ceranae*. PLoS One **6**(6), e21550 (2011)

2. N Simon, Un pas en avant pour la protection contre les pesticides. Abeille et Cie **1**(149), 25–27 (2012)

3. J Blois, Vidéosurveillance d'abeilles, comptage d'entrées/sorties à l'entrée de la ruche. Master's thesis, University of La Rochelle (France) (2011)

4. R Chauvin, Sur la mesure de l'activité des Abeilles au trou de vol d'une ruche a dix cadres. Insectes Sociaux **23**, 75–81 (1976)

5. M Struye, H Mortier, G Arnold, C Miniggio, R Borneck, Microprocessor-controlled monitoring of honeybee flight activity at the hive entrance. Apidologie **25**(4), 384–395 (1994)

6. S Streit, F Bock, CWW Pirk, J Tautz, Automatic life-long monitoring of individual insect behaviour now possible. Zoology (Jena) **106**(3), 169–71 (2003)

7. C Chen, E Yang, J Jiang, T Lin, An imaging system for monitoring the in-and-out activity of honey bees. Comput. Electron. Agric. **89**, 100–109 (2012)

8. T Balch, Z Khan, M Veloso, Automatically tracking and analyzing the behavior of live insect colonies, in *Proceedings of the Fifth International Conference on Autonomous agents*, vol. 2001 (ACM, New York, 2001), pp.521–528

9. J Campbell, L Mummert, R Sukthankar, Video monitoring of honey bee colonies at the hive entrance. Workshop Vis. Observation Anal. Anim. Insect Behav. (ICPR) **8**:1–4 (2008)

10. V Estivill-Castro, D Lattin, F Suraweera, V Vithanage, Tracking bees - a 3d, outdoor small object environment, in *Proceedings of the 2003 International Conference on Image Processing, 2003. ICIP 2003*, (IEEE, Piscataway, 2003), pp.1021–1024

11. B Miranda, J Salas, P Vera, Bumblebees detection and tracking. Workshop Vis. Observation Anal. Anim. Insect Behav. ICPR 2012, (IEEE, Piscataway, 2012)

12. P Viola, M Jones, Rapid object detection using a boosted cascade of simple features, in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001.*, vol. 1, (IEEE, Piscataway, 2001), pp.I-511

13. T Kimura, M Ohashi, R Okada, H Ikeno, A new approach for the simultaneous tracking of multiple honeybees for analysis of hive behavior. Apidologie **42**(5), 607–617 (2011)

14. D Theriault, Z Wu, N Hristov, S Swartz, K Breuer, T Kunz, M Betke, Reconstruction and analysis of 3D trajectories of Brazilian free-tailed bats in flight. Technical report, CS Department, Boston University (2010)

15. Z Khan, T Balch, F Dellaert, A Rao-Blackwellized particle filter for eigentracking, in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004*, vol.2 (IEEE, Piscataway, 2004), pp.II-980

16. A Veeraraghavan, R Chellappa, M Srinivasan, Shape-and-behavior encoded tracking of bee dances. IEEE Trans. Pattern Anal. Mach. Intell. **30**(3), 463–476 (2008)

17. P Maitra, S Schneider, M Shin, Robust bee tracking with adaptive appearance template and geometry-constrained resampling, in *2009 Workshop on Applications of Computer Vision (WACV)*, (IEEE, Piscataway, 2009), pp.1–6

18. C Hendriks, Z Yu, A Lecocq, T Bakker, B Locke, O Terenius, Identifying all individuals in a honeybee hive - progress towards mapping all social interactions. Workshop Vis. Observation Anal. Anim. Insect Behav. ICPR 2012, (IEEE, Piscataway, 2012)

19. Z Khan, T Balch, F Dellaert, Efficient particle filter-based tracking of multiple interacting targets using an MRF-based motion model, in *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS 2003)*, vol.1 (IEEE, Piscataway, 2003), pp.254–259

20. B Ristic, S Arulampalam, N Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. (Artech House, Boston, London, 2004)

21. A Feldman, T Balch, Representing honey bee behavior for recognition using human trainable models. Adaptive Behav. Anim. Animats Softw. Agents Robots Adaptive Syst. **12**(3–4), 241–250 (2004)

22. K Nummiaro, E Koller-meier, T Svoboda, D Roth, LV Gool, Color-based object tracking in multi-camera environments. Lecture Notes in Computer Science **2781**, 591-599 (2003)

23. D Piatti, Time-of-Flight cameras: test, calibration and multi-frame registration for automatic 3D object reconstruction. PhD thesis, Politecnico di Torino, Italy, 2011

24. D Parks, S Fels, Evaluation of background subtraction algorithms with post-processing, in *IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, 2008. AVSS 2008*, (IEEE, Piscataway, 2008), pp.192–199

25. R Kalman, A new approach to linear filtering and prediction problems. J. Basic Eng. **82**, 35–45 (1960)

26. S Blackman, R Popoli, *Design and Analysis of Modern Tracking Systems*, vol. 685. (Artech House, Norwood, 1999)

27. S Julier, J Uhlmann, Unscented filtering and nonlinear estimation. Proc. IEEE **92**(3), 401–422 (2004)

28. C Rasmussen, G Hager, Probabilistic data association methods for tracking complex visual objects. Pattern Anal. Mach. Intell. IEEE Trans. **23**(6), 560–576 (2001)

29. C De Boor, *A Practical Guide to Splines*. (Springer, Heidelberg, 1978)

30. D Reid, An algorithm for tracking multiple targets. Automatic Control IEEE Trans. **24**(6), 843–854 (1979)

31. F Requier, F Brun, P Aupinel, M Henry, JF Odoux, V Bretagnolle, A Decourtye, The composition of agricultural landscapes influences life history traits of honeybee workers, in *European Conference on Behavioural Biology (ECBB VI)*, Essen, Germany (2012)