Constant Angular Velocity Regulation for Visually Guided Terrain Following

Huatian Wang[1], Qinbing Fu[1,3], Hongxin Wang[1], Jigen Peng[2,3], and Shigang Yue[1,3(✉)]

[1] The Computational Intelligence Lab (CIL), School of Computer Science, University of Lincoln, Lincoln LN6 7TS, UK
{hwang,qifu,howang,syue}@lincoln.ac.uk
[2] School of Mathematics and Information Science, Guangzhou University, Guangzhou 510006, China
jgpeng@gzhu.edu.cn
[3] Machine Life and Intelligence Research Center, Guangzhou University, Guangzhou 510006, China

Abstract. Insects use visual cues to control their flight behaviours. By estimating the angular velocity of the visual stimuli and regulating it to a constant value, honeybees can perform a terrain following task which keeps the certain height above the undulated ground. For mimicking this behaviour in a bio-plausible computation structure, this paper presents a new angular velocity decoding model based on the honeybee's behavioural experiments. The model consists of three parts, the texture estimation layer for spatial information extraction, the motion detection layer for temporal information extraction and the decoding layer combining information from pervious layers to estimate the angular velocity. Compared to previous methods on this field, the proposed model produces responses largely independent of the spatial frequency and contrast in grating experiments. The angular velocity based control scheme is proposed to implement the model into a bee simulated by the game engine Unity. The perfect terrain following above patterned ground and successfully flying over irregular textured terrain show its potential for micro unmanned aerial vehicles' terrain following.

Keywords: Insect vision · Flight control · Angular velocity · Terrain following

1 Introduction

The detection of visual motion has been researched for decades to understand how insects, like honeybees and locusts, use visual information to guide their flight behaviours [14,17,19]. Executing a visually guided terrain following is one of the most challenging parts in visual flight control for flying insects. To accomplish this, the insect might either measure the flight speed and adjust the height

accordingly, or estimate the flight height directly. However, it is hard for their tiny brains, with insufficient computation resources, to measure the flight speed and the distance to surface. So a much simpler strategy, holding the angular velocity (the ratio of forward speed and distance) constant, is taken by honeybees for this visual control task [4,16,18].

The angular velocity can be measured by the angular displacement $\Delta\phi$ in a small time interval Δt, that is $\omega = \frac{\Delta\phi}{\Delta t}$. In terrain following scenario, denoting v_x as the forward flight speed and d as the distance to the surface, the angular velocity of the image motion perceived by ventral part of the compound eyes can also be expressed as $\omega = \frac{v_x}{d}$. If the forward speed is maintained by a constant forward thrust, then the flight altitude will change automatically as the distance to ground varies by regulating the angular velocity to a constant value. Problems therefore arise as how insect estimates the angular velocity and keeps it constant.

Biological experiments show that honeybees can fly in the central path of the patterned tunnel by balancing the angular velocity of the image motion on both eyes. What's more, the estimation of the angular velocity is largely independent of the spatial frequency and the contrast [1,18]. Further, the spike recordings also show that the responses of some descending neurons in the honeybee's central nerve cord grow as the angular velocity of the image motion increases [9,10]. Both indicate that honeybees are capable of estimating the angular velocity. However, the neural mechanism underlying this ability is still elusive.

Due to the limitation of the computation ability of insect's tiny brain, the differential techniques, matching or feature-based approaches [3] which are widely used in computer vision are not working here. Hassenstein and Richardt propose a classic elementary motion detecting model describing the mechanism of the motion sensing for animals [7]. The Hassenstein-Reichardt (HR) motion detector produces a much higher response when a progressive motion movement presents (see Fig. 1(a)). HR-balanced detector which has a mirror symmetrical structure with a balance parameter can detect motion from both directions [22] (see Fig. 1(b)).

a HR detector   **b** HR-balanced detector
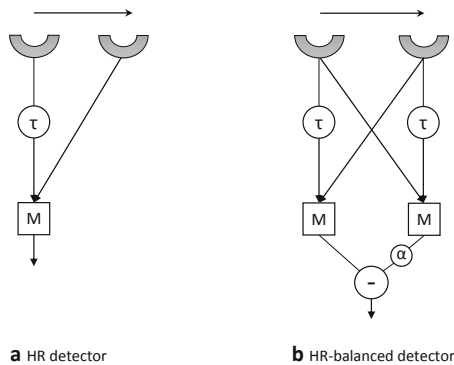
Fig. 1. HR detector and HR-balanced detector. (a) HR detector uses the multiplication (M) of the delayed signal from left photoreceptor and the non-delay signal from right to enhance the response of a preferred direction motion [7]. (b) HR-balanced detector can detect both preferred and opposite direction motions [22].

An angular velocity sensor based on the HR model is designed [12] to accomplish visual guided aircraft terrain following by Franceschini and Ruffier [4,13]. However, both the HR model and the HR-balanced model are tuned for particular temporal frequency (number of gratings passed over the photoreceptor per second) rather than angular velocity [22]. So the output of their sensor shows a large variance when tested by patterned ground in flight [12]. Therefore, an angular velocity specific model is needed to improve the terrain following performance.

According to numerical calculations, Zanker et al. [22] point that the ratio of two HR-balanced detectors is angular velocity tuned. Based on this idea, Cope et al. [2] propose C-HR model, using the ratio of two HR-balanced detectors with different temporal delays. Their model perform well especially around $100°$/s. However, this is different from the fact that the honeybees usually keep a constant angular velocity around $300°$/s [1]. Riabinina and Philippides [11] build up the R-HR model, using a fully temporal dependent channel as the denominator to produce angular velocity tuned responses. But the response of their model slightly depend on the spatial frequency of the moving grating and the dependence increases as the angular velocity grows. Wang et al. [20] propose a model based on the neural structure of Drosophila's visual system which compares the results from detectors with different sampling rates, to get a spatial independent response. Nevertheless, the independence still needs to be more significant before guiding the terrain following.

Previous mentioned models use ratio of two channels to get a angular velocity tuned response, but the spatial independence of the response is not strong enough to reproduce honeybees' flight behaviours. At the same time, the performance is affected when the angular velocity is small or large due to the division. To address this issue, this paper presents a new model using texture estimation layer to estimate the spatial frequency of the image. The proposed model preforms better by combining both texture and temporal information to decode the angular velocity of the image motion.

2   Methods

2.1   Angular Velocity Decoding Model

The model mainly contains 3 parts, the texture estimation part for spatial information extraction, the motion detection part for temporal information extraction and the decoding part for angular velocity estimation. The structure of the model is shown in Fig. 2 explaining how these parts are connected with each other. In the proposed model, every motion detector receives the light intensity change from the neighbouring photoreceptors. And the light intensity change is separated into ON and OFF pathways and then processed by two HR-balanced detectors. The texture information and the motion information from the average of the detectors across the whole vision field are combined. Then angular velocity is decoded from this composite information.
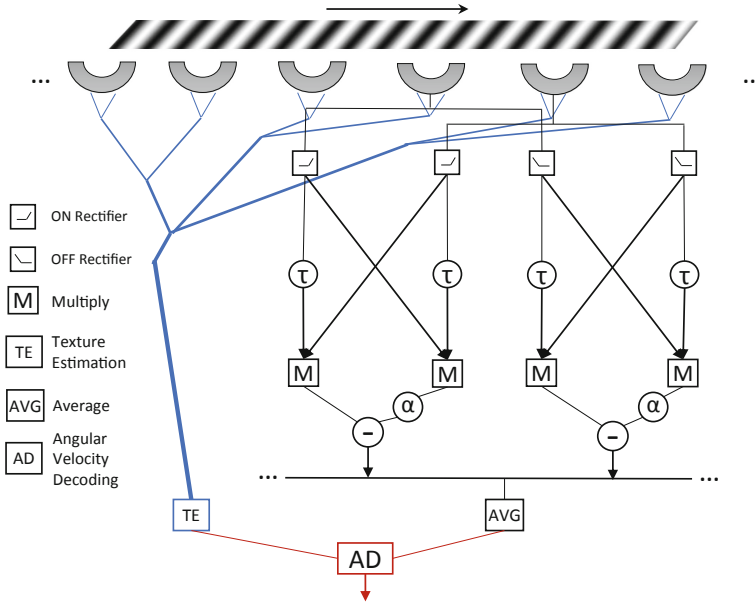
Fig. 2. The proposed angular velocity decoding model. The visual information of the grating movement is received by photoreceptors. The global spatial frequency and image contrast information is estimated by texture estimation part, and the temporal information is processed by motion detectors. Angular velocity decoding layer combines both texture and temporal information to estimate the angular velocity.

(1) Texture Estimation Layer. The simulated input signals received by ventral part of the compound eye are processed by a texture estimation layer where the image contrast and the spatial frequency of the gratings are estimated by the light intensities of different locations. This is based on a hypothesis that insects can have a basic sense of complexity of the texture. And the estimation method requires only low computation ability to give the global texture information of the spatial frequency and image contrast. Here we use Michelson contrast which is defined as the following:

$$C = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} \tag{1}$$

where the I_{max} and I_{min} ($I_{max}, I_{min} \geq 0$) indicate the highest and the lowest light intensities of the input signal in vision filed.

Then in order to decrease the cost of computing, binarization of the input image is performed using the relative intensity threshold $I_{thre} = (I_{max} - I_{min})/2$. The spatial frequency is estimated by counting the number of boundary lines of the binary image in whole visual field. This simple method works well for sine-wave and check-board gratings in our simulations. For more complex background, boundary number also indicates the complexity to some extent.

(2) Motion Detection Layer. Motion detection layer mainly captures the motion information. The input image frames are first processed by the lamina layer where the light intensity change, which insects interest more than the intensity itself, are computed to get the primary information of visual motion. The output of a cell in this layer is defined by the following:

$$P(x,y,t) = I(x,y,t) - I(x,y,t-1) + \sum_{i=1}^{m} p_i P(x,y,t-i) \tag{2}$$

where $P(x,y,t)$ corresponds to the luminance change of pixel (x, y) at time t; m denotes the maximum number of the time steps that the persistence of the luminance change can last and the persistence coefficients $p_i \in (0,1)$ is defined as following equation respectively [21]:

$$p_i = (1 + e^{\mu i})^{-1} \tag{3}$$

Then the luminance changes are separated into two pathways, ON pathway and OFF pathway [5,6]. Specifically, the ON pathway deals with light intensity increments; whilst the OFF pathway processes brightness decrements. Denoting $f^+ = \max(0,f)$ and $f^- = \min(0,f)$, then we can express the outputs of the cells in this two pathways as following:

$$P^{ON}(x,y,t) = P^+(x,y,t), P^{OFF}(x,y,t) = P^-(x,y,t). \tag{4}$$

Considering the delay of the visual signals received by neighbouring cells, we denote $D^{ON}(x,y,t)$ and $D^{OFF}(x,y,t)$ as the output of the ON and OFF detectors for horizontal motion which are computed by following the structure of the HR-balanced detector (Fig. 1(b)). Using a pure time delay of magnitude τ, then we have the following expression:

$$\begin{aligned} D^{ON}(x,y,t) =& P^+(x,y,t-\tau) \cdot P^+(x,y+1,t) \\ &- \alpha P^+(x,y,t) \cdot P^+(x,y+1,t-\tau) \end{aligned} \tag{5}$$

where α is chosen from Zanker's paper [22] setting as 0.25 forming a partial balanced model. And $D^{OFF}(x,y,t)$ can be expressed similarly. Then the output of all motion detectors are averaged to give a response containing the motion information.

(3) Angular Velocity Decoding Layer. In order to decode angular velocity of the image motion from texture information and response of the motion detection layer, here we take only one detector for example to analyse how the response is affected by the input signals. Let S_1, S_2 denote the input signal of photoreceptor A (left) and B (right), and S_1^D, S_2^D denote the temporal delayed signal of A and B, then according to the structure of HR-balanced detector (Fig. 1(b)), the response of the detector R_0 can be expressed as $\overline{S_1^D \cdot S_2 - \alpha S_2^D \cdot S_1}$, where the

bar means the response is averaged over a time period to remove fluctuation caused by oscillatory input.

If the input signals are simulated using the following sinusoidal grating frames (6) with spatial period λ moving at an angular velocity ω:

$$I(x, y, t) = (\sin(\frac{2\pi\omega}{\lambda}(t - \frac{\varphi(y-1)}{\omega})) + 1/C)/(1/C + 1) \tag{6}$$

where (x, y) denotes the location of the ommatidium, t indicates the time and $C \in (0, 1]$ denotes the image contrast. Then we can get the output of each detector by (5). It can be roughly expressed in theoretical [22] as the following equation:

$$R_0 \approx \frac{1 - \alpha}{(1 + C)^2} + \frac{C^2}{2(1 + C)^2}[\sin(\frac{2\pi(\varphi - \tau\omega)}{\lambda}) - \alpha\sin(\frac{2\pi(\varphi + \tau\omega)}{\lambda})]. \tag{7}$$

In fact, the angular velocity of the background moving is caused by the flying of the insects. Considering when a simulated bee performing terrain following, the consistency of the image motion speed in vision field helps us simplify the problem so that we can average the output signals from all ON and OFF detectors in visual field to get the final response $R(\omega, \lambda)$ which encodes the angular velocity.

However, it is hard to derive angular velocity directly from (7). But we can decode the angular velocity information from the response $R(\omega, \lambda)$ using an approximation method. Though there is an inevitable fitting error, we can decrease it into an acceptable level if the fitting function is chosen well. One decoding function can be chosen as following to approximate the actual angular velocity:

$$\widehat{\omega} = a^*\widehat{\lambda}^{b^*}\frac{1 + \widehat{C}}{2\widehat{C}}\sqrt{R} \tag{8}$$

where $\widehat{\omega}$ denotes the decoded angular velocity, $\widehat{\lambda}$ is the estimated spatial period and $\widehat{C}$ is the estimated contrast from texture estimation layer. Parameters a^* and b^* can be learned by minimizing the difference from the ground truth using alternate iteration method:

$$(a^*, b^*) = \arg\min_{a,b} (\omega - a\lambda^b\frac{1 + C}{2C}\sqrt{R(\omega, \lambda)}). \tag{9}$$

2.2   Control Scheme for Automatic Terrain Following

The proposed model can be used to simulate the automatic terrain following of the honeybees by maintaining a constant angular velocity. Using AVDM, we can estimate the angular velocity in flight. By regulating it accordingly to a constant value, the altitude will change automatically even without the exact altitude or forward speed in flight.

The close loop control scheme for terrain following is given in Fig. 3. For simplicity, we assume the forward flight speed is maintained the same by a proper

constant forward thrust. So we only need to adjust the vertical lift according to the difference between the preset angular velocity and the estimated value by AVDM to control the altitude correspondingly. Here, the preset angular velocity is also estimated by the AVDM in the beginning phase when the vertical lift is set to the same value as the gravity and where the ground is flat. After that, when the ventral angular velocity varies caused by terrain undulates, the vertical lift controller will change the lift according to the difference ε between ventral angular velocity estimated and the preset value. If the difference ε is positive, the lift will increase and vice versa.
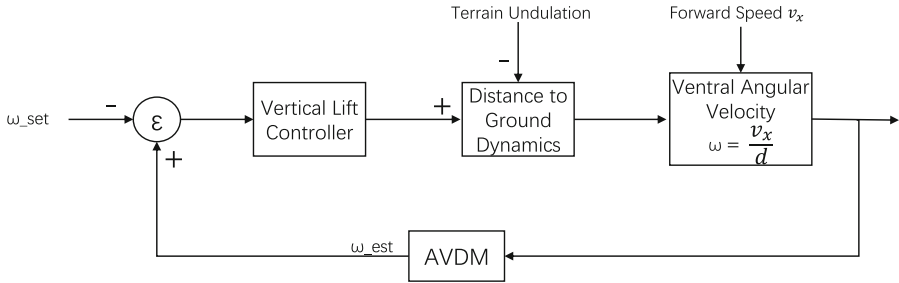
Fig. 3. The AVDM-based close loop control terrain following scheme. The vertical lift controller is triggered by the difference ε between preset angular velocity ω_{set} and the estimated angular velocity ω_{est}.

During the terrain following approach, the vertical speed v_z is relatively small, and the air resistance can be approximated as $f = kv_z$. Then the vertical dynamics can be described using the following differential equations:

$$m\frac{dv_z}{dt} = T - kv_z - mg \tag{10}$$

$$T = \rho(\omega_{est} - \omega_{set}) \tag{11}$$

$$v_z = \frac{dz}{dt} \tag{12}$$

where m is the mass of the simulated bee, g is the gravity acceleration and T is the vertical lift. Given the initial conditions, then the flight trajectory can be computed step by step. In our simulation, this process can be achieved by the physics engine of Unity.

2.3   Parameter Setting

Parameters of the proposed model and the control scheme are shown in Table 1. Parameters are mainly tuned manually based on our empirical knowledge and stay the same in the following simulations unless particularly stated.

**Table 1.** Parameters of the model and the control scheme

Eq.	Parameters
(2)	$m = 10$
(3)	$\mu = 1$
(5)	$\tau = 0.08\,\text{s},\ \alpha = 0.25$
(6)	$\varphi = 2°$
(8)	$a^* = 48.84,\ b^* = 1$
(10)	$k = 0.1,\ g = 9.81$
(11)	$\rho = 0.04$

3 Experiments and Results

Within this section, we present the experiments and results. The proposed model is first tested by synthetic grating stimuli to show its spatial independence in Matlab (© The MathWorks, Inc.). Then the model is implemented into a virtual bee using Unity (© Unity Technologies) to simulate the automatic terrain following behaviours of honeybees.

3.1 Moving Grating Experiments

In the first kind of experiments, we aimed to inspect the spatial frequency independence of the proposed angular velocity decoding model. Therefore, the spatial period of the grating is chosen widely from 12° to 72° and the model is tested by these gratings moving at different angular velocities to get a general result.
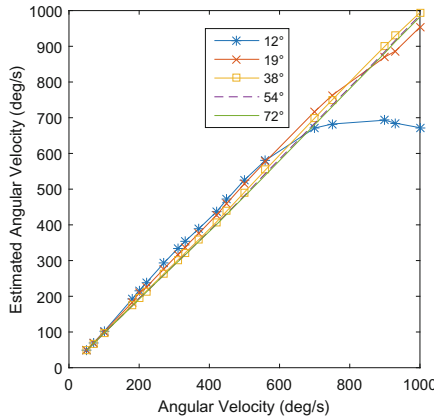
Fig. 4. The estimated angular velocity curves from decoding under different angular velocities when tested by moving gratings of different spatial periods (12°, 19°, 38°, 54° and 72°).

As you can see from Fig. 4, the proposed model show indeed the expected independence of spatial frequency when tested by gratings of different spatial periods. The angular velocities are well decoded with little variance except when the grating is too narrow. This is caused by the much higher temporal frequency when the angular velocity is larger than $700°$/s for grating of $12°$. Actually it does not affect the honeybee's flight in most of the cases since honeybees tend to maintain a constant angular velocity of $300°$/s [1], around which our model shows pretty enough spatial independence.

3.2   Terrain Following Simulations

In the second kind of experiments, the proposed model is implemented in a simulated bee in terrain following simulations where the ground is covered with different textures. The flight trajectories and the ventral responses are recorded to see if the simulated bee can perform automatic terrain following by estimating the angular velocity of the image motion and regulating it to a constant value.

The simulated bee with AVDM implemented is first tested on a regular terrain covered with sinusoidal gratings. The simulated bee is released around a given height at a certain forward speed. In beginning phase, the agent is set to fly forward without changing its altitude (by setting the vertical lift equals gravity), and the preset angular velocity value is estimated using AVDM after the first few frames. Then the control scheme described in previous section starts to take over the control of the vertical lift according to the difference between the angular velocity estimated and the preset value. The result is shown in Fig. 5.
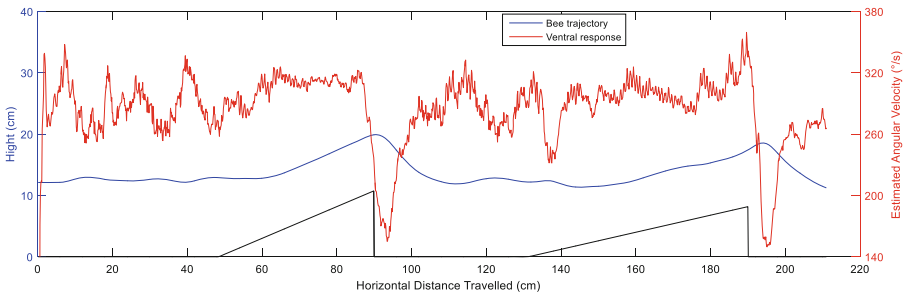
Fig. 5. The bee trajectory (blue line), terrain height (black line) and the angular velocity (red line) estimated by the ventral eye are shown in the same graph. A demo video can be found at https://youtu.be/jaYSuCJGAfc. (Color figure online)

From the flight trajectory, we can see the flight altitude changes automatically to keep a distance from ground using only visual information. This result is similar to the experiments on regular terrain with gratings [12,13]. Using Unity engine, the images received by ventral camera can be processed in real time to estimate the image motion angular velocity. Neither the flight speed nor the flight altitude is necessary to perform this visual guided task.

In order to see whether the model is stable under more complex terrain, we also tested the simulated bee above an irregular mountain terrain covered with sinusoidal gratings. The result is shown in Fig. 6. The flight trajectory of the agent undulates above the mountain shape ground which indicates that the flight altitude changes automatically according to the distance to ground using only visual information. Whenever the distance is close to ground, the increasing angular velocity will trigger the controller to provide a high vertical lift to help the agent get away from ground.
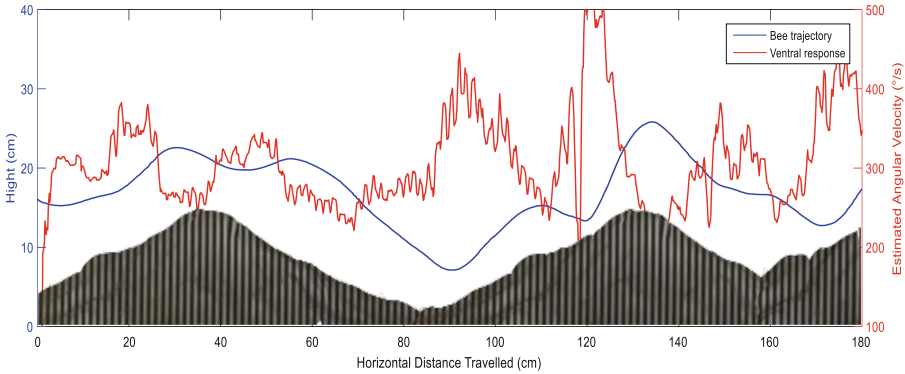

Fig. 6. The bee trajectory (blue line) and the angular velocity (red line) estimated by the ventral eye are shown above the mountain terrain covered with gratings. The simulated bee can adjust its flight height by regulating the angular velocity to a constant value. (Color figure online)

Further, the model is also tested using mountain terrain covered with white snow and black rock in our simulations. In this scenario, the texture information captured by ventral camera is irregular. Especially when there is too much snow in the vision field, the response will drop due to the low contrast. In order to inspect the terrain following ability in this situation, several simulations are performed (see Fig. 7). Though not all flight trajectories follow the terrain well, they all complete the flight tests without crashing. The result is obviously not robust as the terrain following using grating patterns. This is mainly caused by the lack of contrast and in accordance with the phenomenon that bees can plunge straight into calm water where low contrast provided [8].

4   Conclusion and Discussion

We proposed a bio-plausible model, the angular velocity decoding model (AVDM), for estimating the image motion velocity to perform automatic terrain following with only visual information. The model combines both spatial and temporal information from moving frames received by ventral camera to
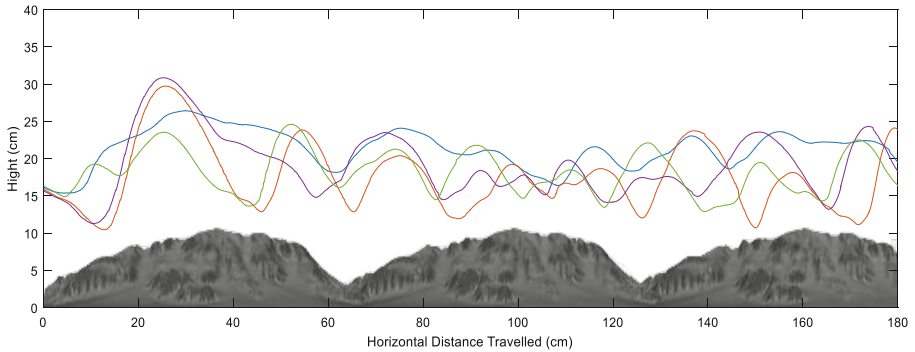
Fig. 7. The simulated bee released around the given height in four times. The flight trajectories are shown above the snow mountain terrain in the same graph. Whenever the distance to ground decreases, the angular velocity increases causing the vertical lift goes up, and so does the flight altitude to help it fly over the terrain.

give a relatively accurate image motion angular velocity. The response curves show large spatial independence in grating simulations which is in accordance with the biological experiments. It also provides a possible explanation of how insects' visual circuits detect image motion speed.

The close loop control scheme based on angular velocity estimation is also proposed to simulate the terrain following of honeybees in game engine Unity. And the result indicates that the model is well capable of guiding the flight course using only visual information. Similar strategy can be used in simulations of the honeybee's wall following behaviour [15]. The control scheme proposed mainly accounts for vertical lift regulating using only ventral part of the panoramic compound eye. In fact, similar strategy can be used to control forward thrust by lateral visual information. Combining both may give a more complete control scheme which can deal with more complex visual flight tasks. The proposed model will be implemented on UAV and be tested in a real environment in the near future.

References

1. Baird, E., Srinivasan, M.V., Zhang, S., Cowling, A.: Visual control of flight speed in honeybees. J. Exp. Biol. **208**(20), 3895–3905 (2005)
2. Cope, A.J., Sabo, C., Gurney, K., Vasilaki, E., Marshall, J.A.: A model for an angular velocity-tuned motion detector accounting for deviations in the corridor-centering response of the bee. PLoS Comput. Biol. **12**(5), e1004887 (2016)
3. Fleet, D.J.: Measurement of Image Velocity, vol. 169. Springer, Heidelberg (2012)

4. Franceschini, N., Ruffier, F., Serres, J.: A bio-inspired flying robot sheds light on insect piloting abilities. Curr. Biol. **17**(4), 329–335 (2007)

5. Fu, Q., Hu, C., Peng, J., Yue, S.: Shaping the collision selectivity in a looming sensitive neuron model with parallel ON and OFF pathways and spike frequency adaptation. Neural Netw. **106**, 127–143 (2018)

6. Fu, Q., Yue, S.: Modeling direction selective visual neural network with on and off pathways for extracting motion cues from cluttered background. In: Proceedings of the IEEE International Joint Conference on Neural Networks, pp. 831–838. Anchorage, AK, USA (2017)

7. Hassenstein, B., Reichardt, W.: Systemtheoretische analyse der zeit-, reihenfolgen- und vorzeichenauswertung bei der bewegungsperzeption des rüsselkäfers chlorophanus. Zeitschrift für Naturforschung B **11**(9–10), 513–524 (1956)

8. Heran, H., Lindauer, M.: Windkompensation und seitenwindkorrektur der bienen beim flug über wasser. Zeitschrift für vergleichende Physiologie **47**(1), 39–55 (1963)

9. Ibbotson, M.: Evidence for velocity-tuned motion-sensitive descending neurons in the honeybee. Proc. Roy. Soc. Lond. B Biol. Sci. **268**(1482), 2195–2201 (2001)

10. Ibbotson, M., Hung, Y.S., Meffin, H., Boeddeker, N., Srinivasan, M.: Neural basis of forward flight control and landing in honeybees. Sci. Rep. **7**(1), 14591 (2017)

11. Riabinina, O., Philippides, A.O.: A model of visual detection of angular speed for bees. J. Theor. Biol. **257**(1), 61–72 (2009)

12. Ruffier, F., Franceschini, N.: Optic flow regulation: the key to aircraft automatic guidance. Robot. Auton. Syst. **50**(4), 177–194 (2005)

13. Ruffier, F., Franceschini, N.: Optic flow regulation in unsteady environments: a tethered MAV achieves terrain following and targeted landing over a moving platform. J. Intell. Robot. Syst. **79**(2), 275–293 (2015)

14. Seidl, R.A.: Die sehfelder und ommatidien divergenzwinkel von Arbeiterin, Königin und drohn der honigbiene (Apis mellifica). Ph.D. thesis (1982)

15. Serres, J.R., Masson, G.P., Ruffier, F., Franceschini, N.: A bee in the corridor: centering and wall-following. Naturwissenschaften **95**(12), 1181 (2008)

16. Serres, J.R., Ruffier, F.: Optic flow-based collision-free strategies: from insects to robots. Arthropod Struct. Dev. **46**(5), 703–717 (2017)

17. Srinivasan, M., Zhang, S., Lehrer, M., Collett, T.: Honeybee navigation en route to the goal: visual flight control and odometry. J. Exp. Biol. **199**(1), 237–244 (1996)

18. Srinivasan, M., Zhang, S.: Visual control of honeybee flight. In: Lehrer, M. (ed.) Orientation and communication in arthropods, pp. 95–113. Springer, Heidelberg (1997). https://doi.org/10.1007/978-3-0348-8878-3_4

19. Wang, H., Peng, J., Yue, S.: A directionally selective small target motion detecting visual neural network in cluttered backgrounds. IEEE Trans. Cybern. (to be published). https://doi.org/10.1109/TCYB20182869384

20. Wang, H., Peng, J., Baxter, P., Zhang, C., Wang, Z., Yue, S.: A model for detection of angular velocity of image motion based on the temporal tuning of the Drosophila. In: Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L., Maglogiannis, I. (eds.) ICANN 2018. LNCS, vol. 11140, pp. 37–46. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01421-6_4

21. Yue, S., Rind, F.C.: Collision detection in complex dynamic scenes using an LGMD-based visual neural network with feature enhancement. IEEE Trans. Neural Netw. **17**(3), 705–716 (2006)

22. Zanker, J.M., Srinivasan, M.V., Egelhaaf, M.: Speed tuning in elementary motion detectors of the correlation type. Biol. Cybern. **80**(2), 109–116 (1999)