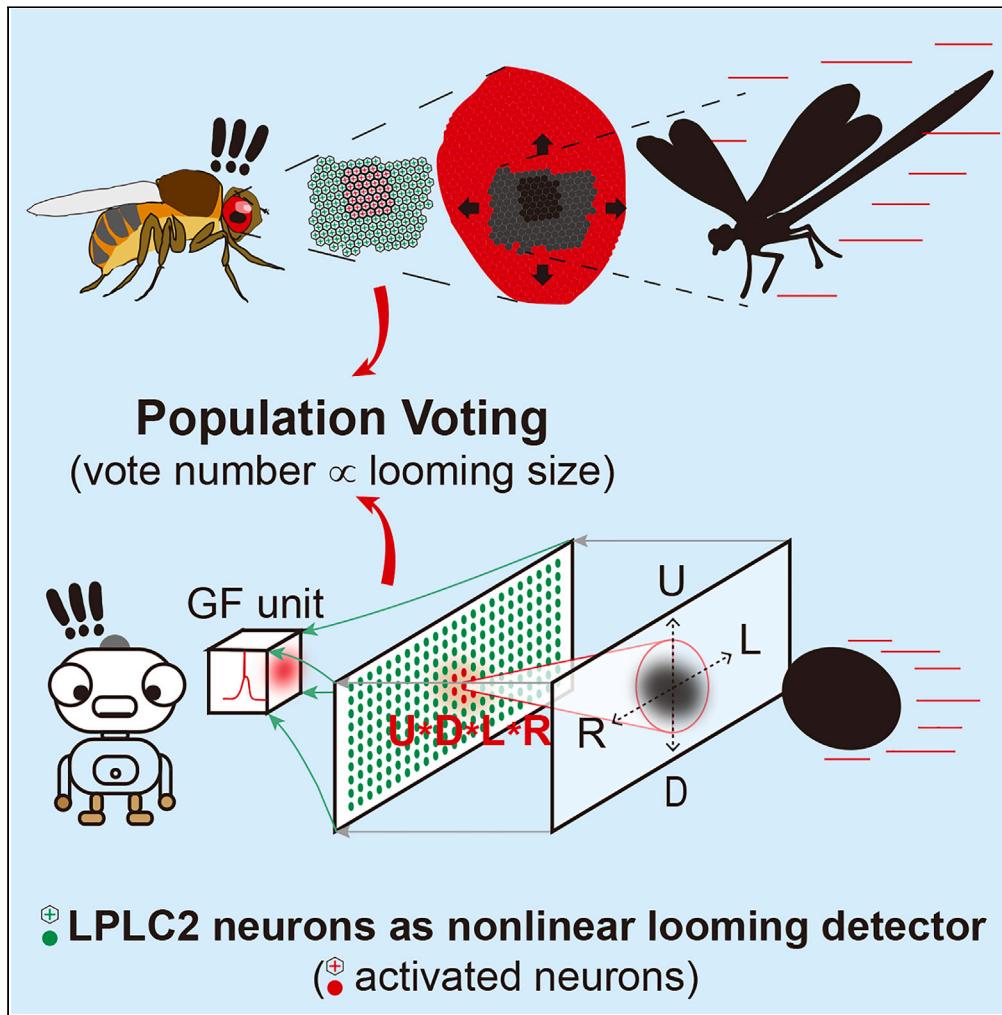


Article

A fly inspired solution to looming detection for collision avoidance



Junyu Zhao,
Shengkai Xi, Yan
Li, Aike Guo,
Zhihua Wu

akguo@shu.edu.cn (A.G.)
wuzh@shu.edu.cn (Z.W.)

Highlights
We model the fast escape circuit of the fruit fly

LPLC2 neurons
nonlinearly integrate EMD
inputs and jointly extract
looming features

Looming-size threshold is
determined by the
receptive field size of
LPLC2 neurons

The fly algorithm enables
robust collision avoidance
in virtual robot tests



Article

A fly inspired solution to looming detection for collision avoidance

Junyu Zhao,^{1,2} Shengkai Xi,^{1,2} Yan Li,^{1,2} Aike Guo,^{1,2,3,4,*} and Zhihua Wu^{1,4,5,*}

SUMMARY

Dodging rapidly approaching objects is a fundamental skill for both animals and intelligent robots. Flies are adept at high-speed collision avoidance. However, it remains unclear whether the fly algorithm can be extracted and is applicable to real-time machine vision. In this study, we developed a computational model inspired by the looming detection circuit recently identified in *Drosophila*. Our results suggest that in the face of considerably noisy local motion signals, the key for the fly circuit to achieve accurate detection is attributed to two computation strategies: population encoding and nonlinear integration. The model is further shown to be an effective algorithm for collision avoidance by virtual robot tests. The algorithm is characterized by practical flexibility, whose looming detection parameters can be modulated depending on factors such as the body size of the robots. The model sheds light on the potential of the concise fly algorithm in real-time applications.

INTRODUCTION

Collision detection ability is of significance to both biological and machine vision systems. Biologically inspired looming detection has long aroused research interest. One recent example is the realization of moving-obstacle detection based on bioinspired vision sensors, i.e., event cameras.¹ Neuromorphic sensors greatly shorten the algorithm's perception latency to moving obstacles. Except for bioinspired event cameras, however, their detection framework was still established upon a traditional approach, such as the use of a Kalman filter and artificial potential field method. Could the solution of biological vision systems for looming detection be accurately extracted and borrowed in real time?

Looming responses are ubiquitous across a wide variety of animal species, including humans, mice, zebrafish, and flies.^{2–6} While neural substrates of looming responses have been extensively identified,^{3,5–13} the underlying neural computation in many species, especially those having visual cortices, is still unclear.

The best-understood circuit mechanisms are from studies on flying insects, which have superb motion vision but tiny and relatively simple brains. In the locust visual system, a highly specialized neuron, i.e., the lobula giant motion detector (LGMD), was identified to be involved in looming detection and related escape behaviors.^{10,14} A number of models have been proposed to elucidate the neural computation underlying LGMD-based looming detection.^{15–18} Another circuit in the *Drosophila* visual system, which connects T4/T5 cells to the giant fiber (GF) descending neuron via lobula plate/lobula columnar, type 2 (LPLC2) visual projection neurons, was recently identified by a series of experimental studies.^{19–22} Differing from LGMD-based circuits that do not receive directionally selective inputs, the fly LPLC2-GF circuit critically depends on the inputs of local motion detectors, i.e., T4/T5 cells.

Different from motion detection techniques used in computer vision such as differential methods and region-based matching methods, and so forth,²³ biological vision uses correlation-based motion detection.^{24–26} A widely accepted model of local motion detectors for biological vision is correlation-type elementary motion detector (EMD), which was formulated more than half a century ago.²⁷ Given that visual motion measurement by the EMD is sensitive to nonmotion-related factors such as luminance contrast and texture properties,^{28,29} it remains unclear how the visual motion-based LPLC2-GF circuit performs accurate looming detection. Another significant question is whether the fly algorithm is applicable to real-time collision avoidance.

In this study, we addressed these issues by developing a computational model constrained by the anatomical properties of the LPLC2-GF circuit. By assuming that individual LPLC2 neurons perform a nonlinear

¹State Key Laboratory of Brain and Cognitive Science, Institute of Biophysics, Chinese Academy of Sciences, Beijing 100101, China

²University of Chinese Academy of Sciences, Beijing 100049, China

³International Academic Center of Complex Systems, Advanced Institute of Natural Sciences, Beijing Normal University at Zhuhai, Zhuhai, Guangdong 519087, China

⁴School of Life Sciences, Shanghai University, Shanghai 200444, China

⁵Lead contact

*Correspondence:
aikguo@shu.edu.cn (A.G.),
wuzh@shu.edu.cn (Z.W.)

<https://doi.org/10.1016/j.isci.2023.106337>



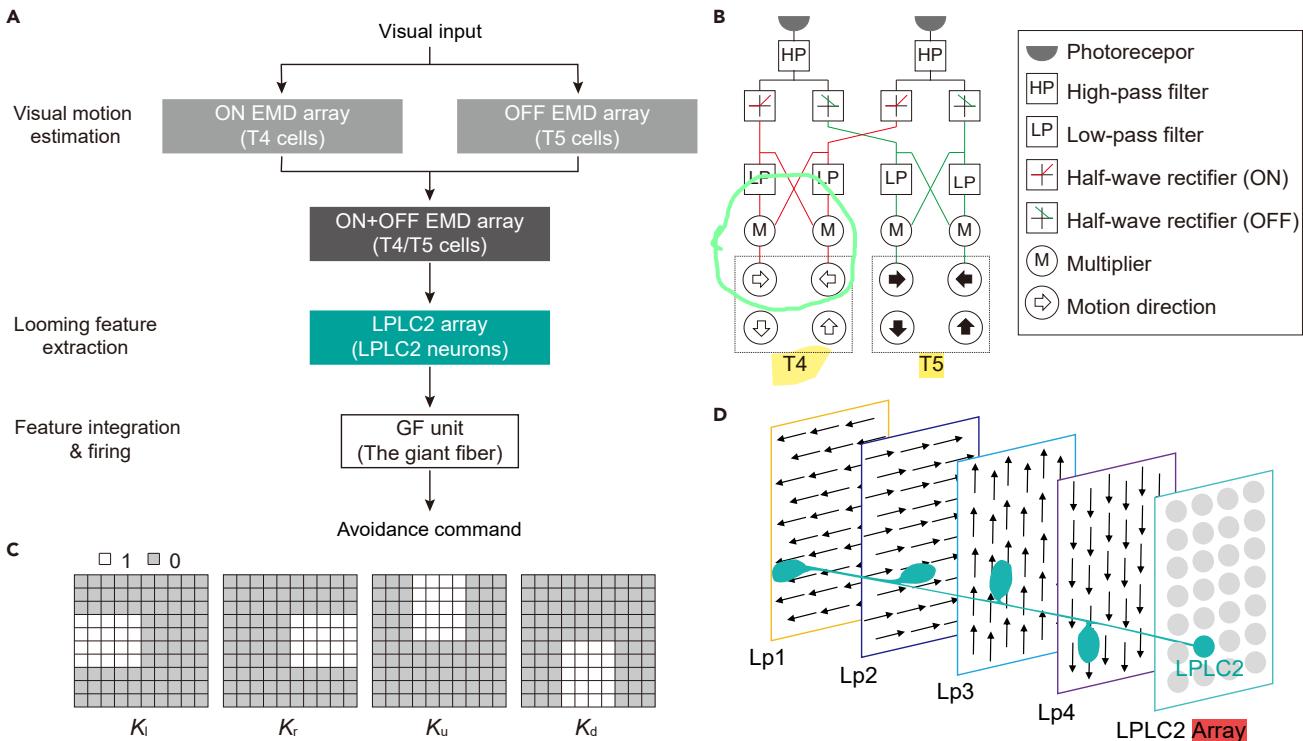


Figure 1. An overview of the looming detection model

(A) Schematic of the EMD-LPLC2-GF network. The biological neurons simulated by the model are denoted by parentheses inside each module, and the simulated function is indicated beside each layer.

(B) Detailed structure of individual **EMD units** (left) with symbolic representations defined (right). The hollow and solid arrows denote the output of motion detectors simulating T4 and T5 cells, respectively.

(C) Four convolution kernels of the LPLC2 layer in (A). Each kernel consists of binary elements responsible for extracting the flow feature in the corresponding cardinal direction. The four kernels together comprise a cross shape, simulating the RF of individual LPLC2 neurons.

(D) Schematic of dendritic arbor spread found in individual LPLC2 neurons in *Drosophila* (Klapoetke et al.²⁰). The representative LPLC2 neuron (green) extends dendrites into four layers of the lobular plate (denoted by LP1, LP2, LP3, and LP4) to extract looming features. Each layer's preferred direction is indicated as an arrow array.

integration of inputs, we show that the model reproduces the physiological response properties of the circuit neurons. The model suggests that the key to fly looming detection lies in **looming feature extraction** by a population coding strategy. The population coding is a means to represent information by using the joint activities of a group of individual neurons.³⁰ Specifically, looming features cannot be restored by independently analyzing the activity of each LPLC2 neuron that responds to the looming stimulus. By contrast, looming features such as the looming size is able to be extracted from the dynamics of the population activity of all LPLC2 neurons that are simultaneously activated by the stimulus, although the responses of the individual LPLC2 neurons themselves could be noisy.

We further tested the model by setting a virtual robot with the task of moving around obstacles. To endow the model with adaptability to robots having a definite body size, we identified a specific parameter in the model that could be adjusted depending on the robot size. Robot simulations show that the model is a robust and effective algorithm for guiding the robot for timely collision avoidance. The superiority of the brain-inspired algorithm over optic flow-based algorithms such as the flow divergence algorithm is examined in the discussion. To the best of our knowledge, the model is the first fly inspired algorithm that can be applicable to real-time collision avoidance.

RESULTS

Nonlinear integration causes LPLC2 units to selectively respond to radial expansion stimuli

Inspired by the circuit connecting T4/T5 cells to single GF descending neuron via LPLC2 neurons in *Drosophila*,^{19,20} we developed a three-layered feedforward network model (Figure 1A). The first layer is an

array of EMD units simulating T4/T5 cells, which estimate visual motion based on visual inputs. Detailed structure of individual EMD units (Figure 1B) was developed based on the model proposed by Eichner et al.,³¹ which reproduced a variety of experimental datasets (Figure 4A in Eichner et al.,³¹) (see STAR Methods). Significant progress has been made in models of the fly EMD over the past decade.³² Individual EMDs have been revealed to be three-armed rather than two-armed in both the ON and the OFF motion pathway in *Drosophila*.^{33,34} We chose, however, to retain the two-armed EMD structure in the present study. The reason is that the two-armed Hassenstein-Reichardt model yields a high degree of directional selectivity that is characteristic of a three-armed EMD model. Compared with the three-armed EMD model, however, two-armed EMD models have been extensively studied. Their pattern-dependent response properties have been clearly known (Buchner 1984; Reichardt 1987)^{28,29}, which would greatly simplify and facilitate the present study.

The second layer is an array of LPLC2 units simulating LPLC2 neurons, which extract looming features based on visual motion. The last layer consists of a GF unit simulating the single GF descending neuron, which integrates the approaching features of the looming object. The model's working hypothesis is that individual LPLC2 units nonlinearly integrate visual motion expanding in the four cardinal directions multiplicatively (see Equation 5 in STAR Methods).

For a given LPLC2 unit, its receptive field (RF) consists of four layers, in each of which the sub-RF forms one arm of a cross shape (Figure 1C). The four-layered RF qualitatively mimics the four-lobe structure found in the RF of individual LPLC2 neurons²⁰ (Figure 1D). According to the model (Equations 1–4 in STAR Methods), motion in a preferred and null direction in each of the sub-RFs gives excitatory and inhibitory inputs to the LPLC2 unit, respectively. The inputs in the four sub-RFs are finally combined through a multiplication operation (Equation 5 in STAR Methods), causing the LPLC2 unit to activate only if it detects preferred direction motion simultaneously in all four cardinal directions from the upstream EMD array. In other words, as local looming detectors, individual LPLC2 units should remain silent when facing nonexpansive stimuli such as unidirectional movement or radial contraction motion.

To test the above assumption, we reproduced a variety of visual stimuli used in a previous study²⁰ (Figure 2A). Among these stimuli, only the one shown in the first column and first row in Figure 2A is a looming stimulus (see STAR Methods). The expanding square with a constant edge speed is not a looming stimulus (Figure 2A, the panel in the second column and first row). The parameter L/v , i.e., a size-to-velocity ratio was used to characterize a looming object's approaching kinematics.¹⁶ The object's half-size is L , and its approaching velocity is v (Figure 2C).

We fed the stimuli to the model and recorded the response of the downstream LPLC2 unit that was retinotopically aligned with the focus of expansion (FOE) motion (denoted as unit O below). The results show that although the EMD array was activated by each type of stimulus, unit O remained silent except for three radial expansion stimuli (Figure 2A, second row). The model responses to the looming square (first col.), the expanding square (second col.), a variety of unidirectional motion (third to eighth col.), and outward (ninth col.) or inward (10th col.) cross-motion are qualitatively consistent with the essential responses of LPLC2 neurons shown in Figure 2D (upper panel) in Klapoetke et al.,²⁰ in the first panel of Figure 2F in Klapoetke et al.,²⁰ in the second to seventh panel of Figure 2F in Klapoetke et al.,²⁰ in the panel b and c of Figure 4A in Klapoetke et al.,²⁰ respectively.

The results were qualitatively unchanged if we fed motion-defined stimuli, whose texture consists of randomly distributed black and white dots and thus makes the foreground figure invisible unless it is moving relative to the background (Figure 2B, first row: motion-defined stimuli; second row: responses of the unit O).

To probe whether a multiplicative integration of EMD inputs is critical for the selectivity of individual LPLC2 units to radial expansion stimuli, we replaced the multiplicative operation in Equation 5 by a summation operation. Specifically, the state of individual LPLC2 units was simulated as an additive integration of visual motion in the four cardinal directions in their RFs, i.e., $LPLC2_{ij}(t) = [L_{ij}^1(t) - L_0]^+ + [L_{ij}^2(t) - L_0]^+ + [L_{ij}^3(t) - L_0]^+ + [L_{ij}^4(t) - L_1]^+$. The function $[x]^+$ is defined as $\max(x, 0)$, and L_{ij}^K indicates visual motion in the cardinal direction K (see STAR Methods for detailed parameter definition). Repeating the above simulations, we found that the LPLC2 unit O responded to nearly all of the stimuli, including wide-field grating motion and unidirectional motion of a bar or edge (Figure 2A, third row). The results were qualitatively unchanged if the stimuli were replaced by motion-defined stimuli (Figure 2B,

与FOE
对齐的
LPLC2
unit

column 1, 2, 9.

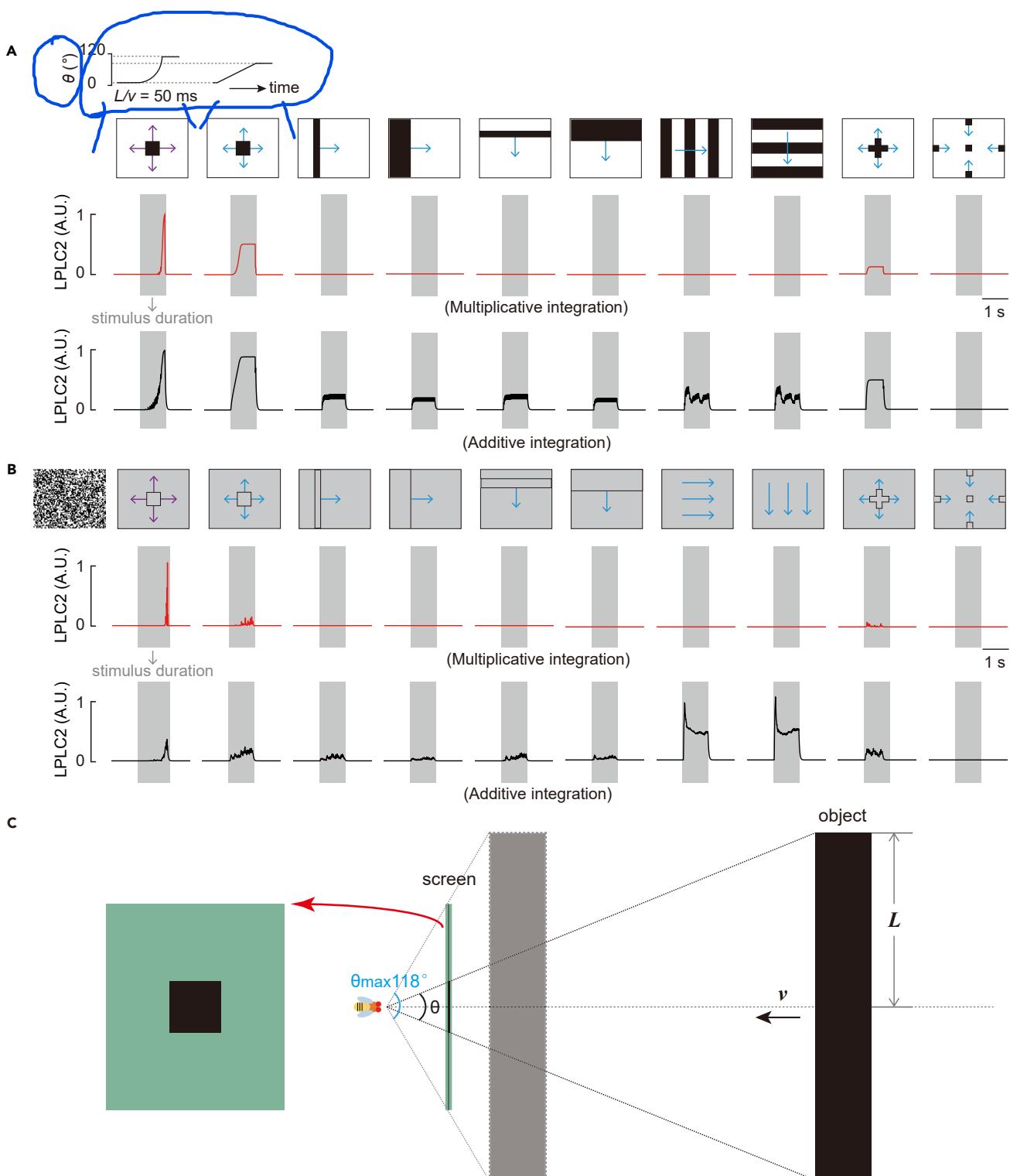


Figure 2. Response properties of the LPLC2 model units

(A) Responses of the centrally located unit in the LPLC2 layer to a variety of motion stimuli, including outward square motion (first and second col.), unidirectional motion (a bar: third and fifth col.; an edge: fourth and sixth col.; a square-wave grating: seventh and eighth col.), and outward (ninth col.) or inward (10th col.) cross motion. It is worth noting that the outward motion in the first col. and first row is a looming stimulus with $L/v = 50$ ms, whereas the other outward motion (in the second and ninth col. and first row) is not looming but expanding with a constant edge speed, as illustrated by their temporal characteristics (two panels at the top of the first and second col.). All the stimuli had a constant edge velocity of 50 pixels per second except for the looming

Figure 2. Continued

one. Simulation parameters are the same for the results in the second and third row except for the integration operation by the **LPLC2 units**, as indicated under each row.

(B) Same as (A) except that each stimulus in (A) was replaced by a corresponding **motion-defined stimulus**, whose texture consists of randomly distributed **black and white dots** (leftmost panel in the first row). **The foreground figure is invisible unless it is moving relative to the background.** The texture in all the motion-defined stimulus is represented by a uniform gray color for drawing clarity. The foreground is outlined by straight lines, and the square-wave grating is represented by three parallel arrows for drawing clarity.

(C) **Top view** diagram illustrating the presentation of looming stimuli. The approaching square objects (half-size, L in both horizontal and vertical directions; velocity, v) projected on the screen were fed into the network model (model input, black square on the green area), giving the model a $118^\circ \times 103^\circ$ -field of view. θ is the object's horizontal angle subtended to the modeled eye, whose maximal value (denoted as θ_{\max}) is reached when the object moves to the location marked by the gray area.

third row). These results indicate that **nonlinear integration** makes LPLC2 **units** selectively respond to radial expansion stimuli.

Looming features are extracted in the LPLC2 layer by a population coding strategy

Our simulations found that looming stimuli always induced **population activity** in the LPLC2 **array**, whose central unit O retinotopically pointed to the FOE (Figure 3A, panel A7). The activity spatial scope rapidly reached the boundary of an inner area within the RF of unit O , after which the activation began to retract and then vanished (see *Video S1*). **The inner area was found to have a side length that is approximately half the side length of the RF.** The underlying reason was that individual LPLC2 units were activated only if four arms of their RFs were simultaneously swept by expanding visual motion (Figure 3A). Even if three of the four arms of an RF were swept by expanding visual motion, the corresponding unit could not be activated (Figure 3A, panel A4).

We examined the number of instantaneously activated LPLC2 units (denoted as $N_{act}(t)$ below) in the population. The number showed a negatively skewed unimodal curve over time (Figure 3B). An obvious question to ask is whether the spatially limited population activity encoded approaching features of the looming object.

We created a set of looming stimuli, whose parameter L/v fell within the range [10 ms, 100 ms]. The looming object's subtended angle, measured by both plane angle (Figure 3C) and solid angle (Figure 3D), rapidly increased with time on its collision course. By feeding each of the stimuli to the model, our simulations showed that within the rising part of the $N_{act}(t)$ curve, the number of instantaneously activated LPLC2 units increased with time in a trend similar to the looming size on visual inspection (Figure 3E vs. Figures 3C and 3D). The rising part of the unimodal curve shifted each other along the time axis in an L/v value-dependent way. The larger the L/v value was, the earlier the curve peak was.

We further plotted $N_{act}(t)$ against the looming size measured by plane angle (Figure 3F) and solid angle (Figure 3G). Both plots confirmed that within the rising part of the curves, the number of instantaneously activated LPLC2 units increased with the looming size across all the L/v values tested. Compared with the plane angle, the solid angle was almost perfectly linearly related to $N_{act}(t)$, especially for $N_{act}(t) < 1,000$. In summary, these results indicate that the looming size was well extracted by the LPLC2 units in a population coding strategy.

The giant fiber unit normalizes various looming stimuli into its firing response occurring at a consistent angular-size threshold

Driven by presynaptic LPLC2 units, the GF unit was depolarized to spike firing during looming, as illustrated by an example in Figure 4A. To examine the relationship between the GF peak responses and approaching kinematics, we plotted the GF peak firing rate (Figure 4B, blue dots), the corresponding peak time (Figure 4C, blue dots), and peak looming size (Figure 4D, blue dots) against the L/v value. The results showed that the GF peak firing rate monotonically decreased with the L/v value except for an **abnormal elevation at medium L/v (50–60 ms)**. The peak firing time was almost linearly related to the L/v ratio and corresponded to a particular looming size. In other words, the GF peak firing occurred at a consistent angular-size threshold that was **independent of the L/v value**. The specific angular-size threshold was consistently at approximately $\theta_{peak} = 55^\circ$ across all the L/v values tested.

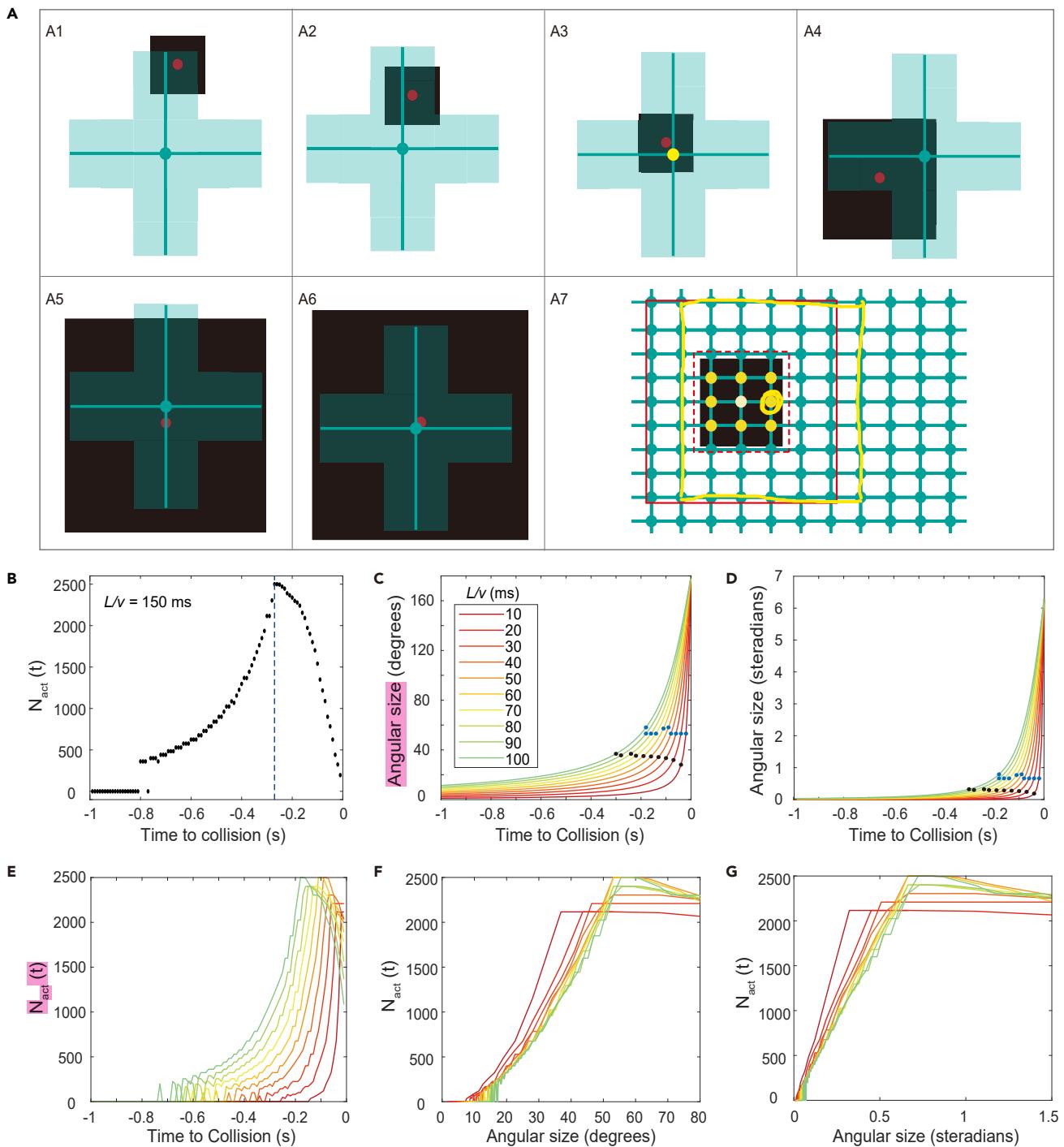


Figure 3. LPLC2 population coding

(A) Schematic of the activation state of representative LPLC2 units (A1–A6) and the LPLC2 array (A7) under a looming stimulus. The black square denotes an instantaneous snapshot of the looming object projected on the modeled eye, whose FOE is marked by a red dot. Individual LPLC2 units (green dots at cross centers) were activated (yellow dot) only if their RFs (light green area in A1–A6) were swept by expanding visual motion simultaneously in the four cardinal directions. Consequently, a population of LPLC2 units constrained only within a limited area (dotted boundary) could be activated (yellow dot and dark yellow dots) on the collision course of the looming object (A7) (see also Video S1). The limited area has a side length of half of the RF side. The RF of the center unit of the activated population is marked by the red boundary (A7).

Figure 3. Continued

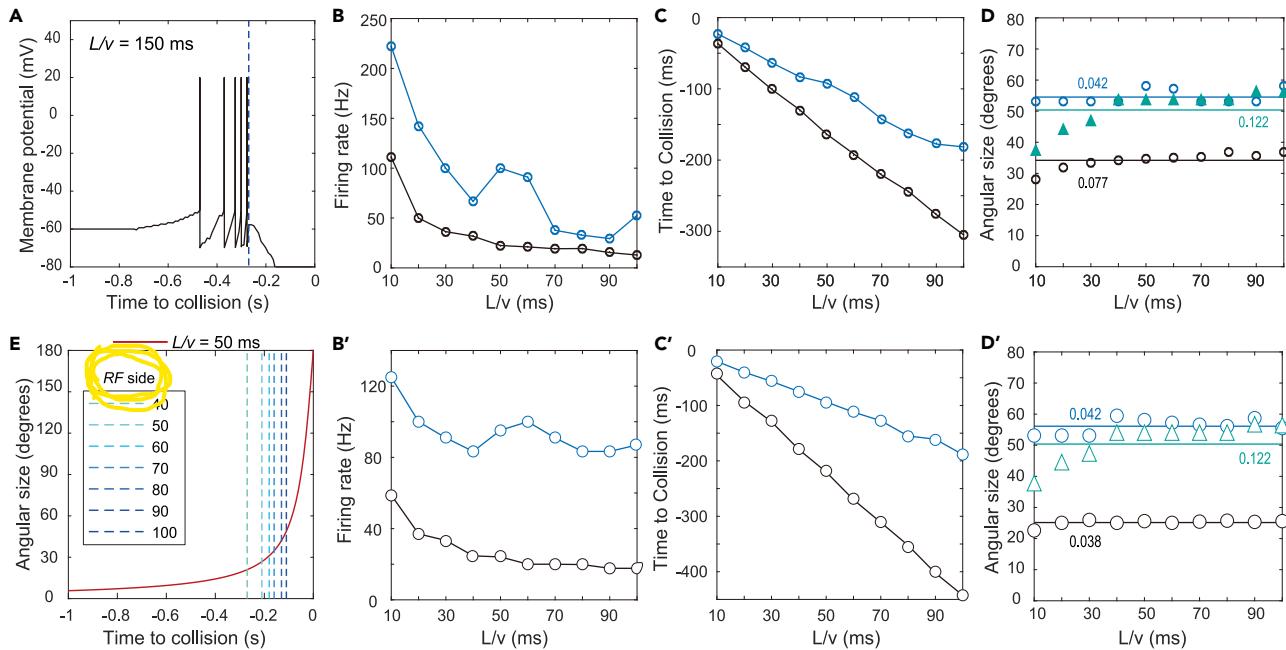
(B) Number of instantaneously activated LPLC2 units (denoted as $N_{act}(t)$) induced by a looming stimulus with the L/v value indicated in the panel. Peak time is marked by the dashed line. On the x axis, $t = 0$ represents the time of expected collision when the looming size reaches 180° and $t < 0$ before collision.

(C and D) Temporal characteristics of the angular size subtended by the looming object and the GF firing under different L/v conditions. The looming size was separately measured by plane angle (C) and solid angle (D). The black and blue solid circles show the first spike time and the peak response time of the GF unit, respectively. The L/v value of the stimulus was systematically varied as specified in the panel.

(E) The number $N_{act}(t)$ of instantaneously activated LPLC2 units versus time under each stimulus given in (C).

(F and G) Same data as in (E) except for being replotted to display a relationship between $N_{act}(t)$ and looming size. $N_{act}(t)$ is more linearly correlated with the solid angle (G), although $N_{act}(t)$ is also positively correlated with the plane angle (F).

The linear dependence of the GF peak firing time on the L/v ratio (Figure 4C, blue dots) is qualitatively consistent with the linear relationship between the latency of the GF peak response and L/v (e.g., Figure 8B (black squares) in von Reyn et al.²²). Except for the peak time, a direct comparison of the GF unit's spiking in our model (e.g., the example in Figure 4A) with the GF response data is irrational. This is because looming stimuli usually induced a gradual GF depolarization but not GF spikes in most flies.^{21,22} The GF response under the condition of LC4 silencing should be, however, comparable to the number of instantaneously activated LPLC2 units $N_{act}(t)$ in our model, because the remaining GF responses after LC4 silencing derive from LPLC2 input.¹⁹ The unimodal curve $N_{act}(t)$ in our model (e.g., Figure 3B) is qualitatively consistent with the non-LC4 component of the GF response (red curves in Figure 7A in von Reyn et al.²²). The rising part of the $N_{act}(t)$ curve vs. the looming object's subtended angle in our model (Figure 3F) is qualitatively consistent with the rising part of the non-LC4 component of the GF response plotted with respect to the instantaneous angular size (Figure 7C in von Reyn et al.²²). These aligned properties of our results

**Figure 4. Emergence of an angular-size threshold in the GF unit**

(A) The simulated membrane potential of the GF unit simultaneously recorded with the data in Figure 3B. The dashed line marks the peak time of $N_{act}(t)$. (B) Instantaneous firing rates of the GF unit across looming L/v values at the time of its first spike (black solid circles) and the time of its peak response (blue solid circles).

(C) The first spike time (black solid circles) and the peak response time (blue solid circles) of the GF unit across looming L/v values.

(D) Instantaneous looming size when the GF unit fired its first spike (black solid circles) or reached peak firing (blue solid circles) and when the $N_{act}(t)$ peaked (green triangles) across looming L/v values. The color-coded lines denote the average values for each sample. The color-coded numbers beside the lines indicate the corresponding coefficient of variation.

(E) The peak time of $N_{act}(t)$ (dashed lines) under different conditions of the RF side length of LPLC2 units. The same looming stimulus (L/v value, at the top of the panel) was used for all the simulations, whose temporal characteristics of the looming size are displayed (red curve).

(B', C', D') Same as (B), (C), and (D), respectively, except that the multiplicative integration of the looming size and velocity information in the GF unit was replaced by a summation operation. The circles represent the same meaning as the solid circles with the corresponding color between each pair of panels.

with the observation of looming responses of the GF cell indicate that our model qualitatively mimics the LPLC2–GF circuit at the algorithmic level. Meanwhile, the properties (such as the **consistent angular-size threshold** that was **independent of the L/v value**) are also consistent with findings in locust¹⁶ and zebrafish larvae,⁵ indicating that different animal species share a similar neural mechanism.

An important question is whether the angular-size threshold is totally determined by the $N_{act}(t)$ input. We then calculated the **coefficient of variation (CV)** of two samples: the **looming size when $N_{act}(t)$ peaked** (**Figure 4D**, green triangles) and the **looming size when the GF peak firing occurred** (**Figure 4D**, blue dots). The analysis showed that the **CV of the looming size corresponding to the $N_{act}(t)$ peaks** was approximately **3 times as large as that of the looming size corresponding to the GF peak firing** (the numbers in **Figure 4D**). The results indicate that the **looming velocity input** ($\Delta N_{act}(t)/\Delta t$ in **Equation 7**) makes an indispensable contribution to making the **angular-size threshold consistent** across various L/v values, especially at small L/v (fast looms), although the **$N_{act}(t)$ input** plays an essential role in the emergence of the angular-size threshold.

Since a neuron's first spike represents a significant increase in the degree of its depolarization, we thus checked how the first spike changes with the L/v value. The simultaneously recorded data showed that the GF firing rate around its first spike monotonically decreased with the L/v value (**Figure 4B**, black dots). The first spike time of the GF unit was linearly related to the L/v ratio as was its peak firing time (**Figure 4C**, black dots). The first spike also occurred at a consistent angular-size threshold $\theta_{first-spoke} = 34^\circ$ which was independent of the L/v value (**Figure 4D**, black dots). Considering that the **GF first spike** provided an earlier time for looming detection than its **peak firing** (black versus blue dots in **Figures 3C**, **3D**, and **4C**), we chose the former rather than the latter as **the avoidance time in virtual robot simulations below**.

By further changing the RF of the LPLC2 units, our simulations showed that the peak time of the $N_{act}(t)$ curve was determined by the RF size (**Figure 4E**, dashed lines). **The smaller the RF is, the earlier the $N_{act}(t)$ peak is, and the smaller the angular-size thresholds ($\theta_{first-spoke}$ and θ_{peak}) should be.** The RF was set as a cross in a square of **100 × 100 pixels** (i.e., **80° × 80°**) in our simulations, unless otherwise specified. This RF size is larger than the real size of **60° × 60°** measured in LPLC2 neurons,²⁰ which was chosen to ensure a reasonable $\theta_{first-spoke}$ in our model. The curves of a looming object's subtended angular size versus time could be split into three phases: early, middle, and late (**Figures 3C** and **3D**). Compared with the other two, the middle phase, approximately falling within 20° – 50° , shows the largest sensitivity to different L/v conditions on visual inspection. The $\theta_{first-spoke} = 34^\circ$ (**Figure 3C**, black dots) fell within the phase having the largest sensitivity.

To examine whether a multiplicative integration of looming size and velocity in the GF unit is critical for its firing properties above, we replaced the multiplicative operation in **Equation 7** by a summation operation. Repeating the above simulations, the results showed that the GF firing properties were qualitatively the same as those obtained above (**Figures 4B'**, **4C'**, **4D'**), indicating that the emergence of the angular-size thresholds was not affected by the specific integration operation of looming size and velocity in the GF unit.

Taken together, the postsynaptic GF unit normalized the approaching kinematics with various L/v values into its **first spike firing** or its **peak firing** that occurred at a **consistent angular-size threshold**, to which not only the **looming size** but also the **looming velocity** played an indispensable role.

Looming detection under moving background conditions

All looming stimuli in the above simulations feature a static background. It is necessary to ask whether the model works when the background is not stationary as in the real-world. To this end, either a translational natural scene (**Figure 5A**) or a scene captured by a driving recorder mounted on a vehicle moving forward³⁵ (**Figure 5B**) was used as a background. By superimposing a dark or bright looming square with various L/v values upon each of these backgrounds, we synthesized image sequences or Video clips as new stimuli (see **Videos S2** and **S3**).

By presenting the synthetic stimuli to the model one by one, our simulations show that for nearly all of the stimuli tested, the looming object was successfully detected, which was indicated by the appearance of spike firing in the GF unit (**Figures 5A** and **5B**). Even if the background was slowly expanding, the model capably detected the looming foreground object (**Figure 5B**). A few exceptions occurred for the stimuli

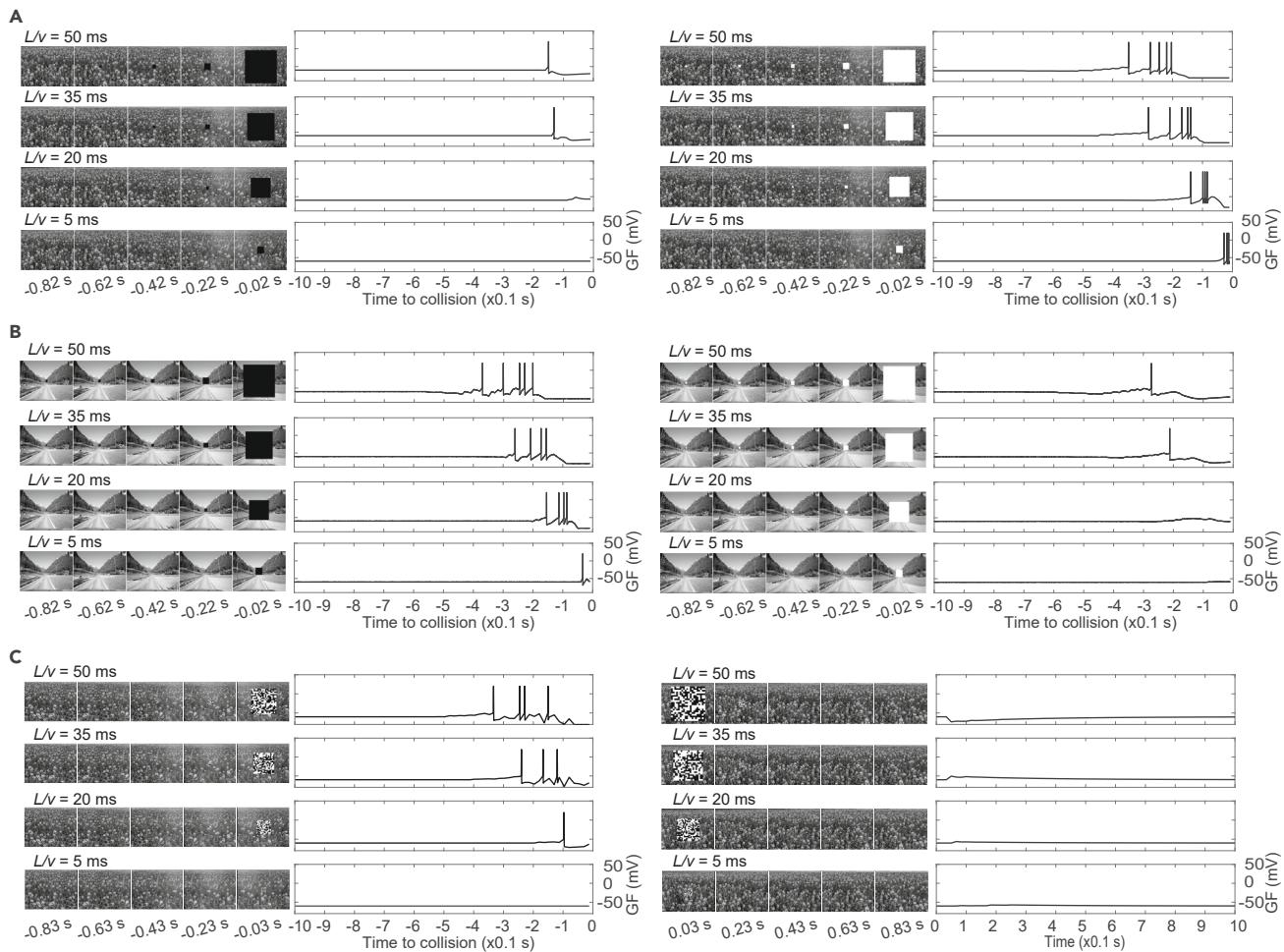


Figure 5. Looming detection by the model is robust to moving background and object texture

(A) The simulated **membrane potential of the GF unit** (second and fourth col.) on a direct collision course of a dark (first col.) or bright (third col.) looming square. The background of the stimulus was moving leftwards at a constant speed of 1200 pixels per second (see [Video S2](#)). The frame images were preprocessed to a size of $N \times M = 400 \times 300$ pixels by [downsampling](#). Representative frames at the time stamp marked are displayed, at the top of which the corresponding L/v value is specified.

(B) The simulated membrane potential of the GF unit (second and fourth col.) on a direct collision course of a dark (first col.) or bright (third col.) looming square. The expanding background Video was taken by a driving recorder mounted on a vehicle moving forward³⁵ (see [Video S3](#)).

(C) The simulated membrane potential of the GF unit (second and fourth col.) on a direct collision (first col.) or receding (third col.) course of a textured square consisting of randomly distributed black and white dots. The background is the same as used in (A) except that its speed is 1,440 pixels per second (see [Video S4](#)). The GF unit remained silent during the receding course as predicted by the model.

that were not only very fast but also had a low contrast, such as the case with a dark looming object with $L/v = 5$ ms in [Figure 5A](#).

Does the model only detect objects with a solid texture from a moving natural scene? We next created a textured looming square consisting of randomly distributed black and white dots. By superimposing the square with various L/v values upon a translational natural scene, we synthesized a set of new stimuli (e.g., [Video S4](#), the left side of the screen). By presenting the synthetic stimuli to the model one by one, our simulations show that the textured object was successfully detected in most cases, except for the condition of very fast looming stimuli ([Figure 5C](#), fourth row in the left half). To further confirm that the model output should remain silent when facing a receding textured object, we replaced the textured square by the same but receding one in each of the stimuli above (e.g., [Video S4](#), the right side of the screen). The repeated simulations show that the LPC2 units did not respond to the receding textured object as expected, which was indicated by the inactivated or silent GF unit ([Figure 5C](#), the right half).

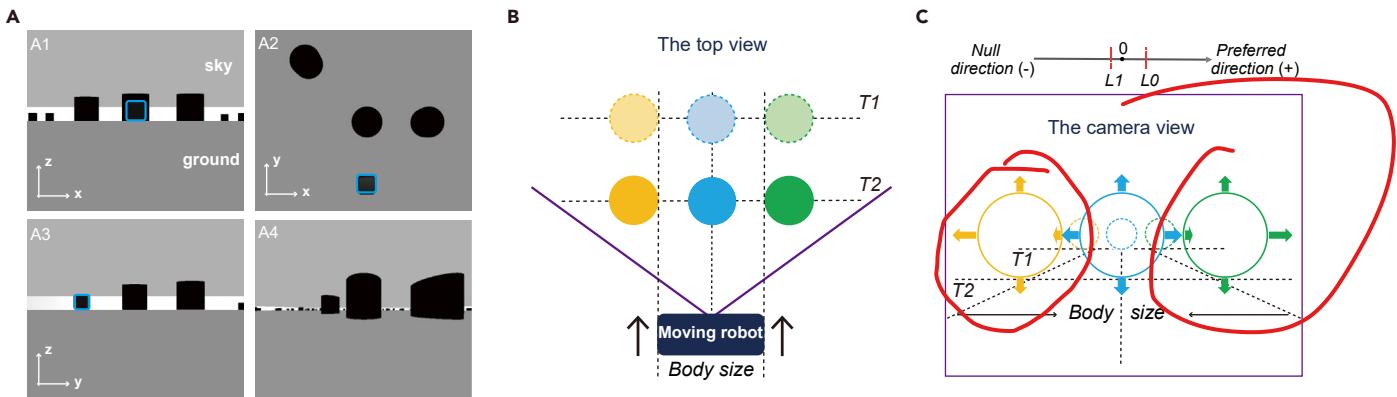


Figure 6. Closed-loop tests by applying the model to a virtual robot

(A) Simulation arena viewed separately from the positive y axis (A1), negative z axis (A2), negative x axis (A3), and from a camera mounted on the robot (A4). The robot (marked by a blue boundary) saw its environment (A4) by receiving a real-time Video stream from the camera. Frame images were preprocessed to $N \times M = 300 \times 250$ pixels by downsampling. Randomly placed cylindrical objects were used as obstacles.

(B) Top view of three stationary obstacles (orange, blue, and green circles) experienced by the moving robot (arrows: moving direction). The obstacles were arranged along a line (horizontal dashed lines) perpendicular to the robot's heading direction (vertical dashed lines). The obstacles at time T_2 (solid circles) were nearer to the robot than at an earlier time T_1 (dashed circles). Vertical lines illustrate that the robot will be next to but not touch the orange and green objects.

(C) The same scene as that in (B) except that the view is from the robot camera itself. From the robot's field of view (purple boundary), the obstacle sizes changed from small (dashed circles) at time T_1 to large (solid circles) at time T_2 . If the robot was a mass point, only the blue object that induced expanding optic flow in all four cardinal directions (blue arrows) was detected as a looming object. However, it had a definite body size and thus would probably collide with the orange and green objects. The threshold L_1 in Equation 5 should be tuned as negative (top), so that the model can detect obstacles whose optic flow amplitudes are larger than L_0 in three cardinal directions and simultaneously larger than L_1 in another direction.

In short, the model is qualified for detecting a looming object from a moving natural scene background.

Collision avoidance in a virtual robot based on estimated visual motion

Compared with the open-loop conditions above, looming detection is much more challenging in real physical scenes because of a critical requirement for collision avoidance. Collision avoidance actions bring a rapid and sudden change in visual inputs to the agent, making real-time looming detection harder. By equipping TurtleBot 3.0 robots with the model (see STAR Methods), we investigated whether the model could be an efficient algorithm for real-time collision avoidance.

The robot saw surrounding obstacles within its 120-degree field of view through a camera mounted on it (Figure 6A). We first take an example of the robot that kept moving straight ahead toward three stationary obstacles (Figures 6B and 6C). In addition to obstacles straight ahead (blue object in Figures 6B and 6C), objects on front-left or front-right, which induce optic flow expanding in three cardinal directions but slightly contracting in the fourth cardinal direction (orange and green objects in Figures 6B and 6C), probably collided with the moving robot. This is because the robot is not an ideal mass point but has a definite body size. To solve the problem, we proposed that the parameter L_1 in the state function of LPLC2 units should be negative (Equation 5), whose specific value should depend on the robot's body size and moving speed (Figure 6C). In our simulations with constant robot speed and body size below, the values of parameters L_1 and L_0 were kept unchanged (Table 1). Therefore, in closed-loop conditions, the model operated in the same way as in open-loop conditions except for a two-step flow detection. First, the LPLC2 layer kept detecting a flow pattern that expanded simultaneously in any three cardinal directions. Once such a flow pattern was caught, the layer then checked whether the pattern satisfied the condition $[L_{ij}^4(t) - L_1]^+ > 0$ in another cardinal direction (Equation 5).

Once an obstacle was detected, the model triggered a control module by giving the robot an avoidance command with two messages. One was the avoidance time signaled by the first spike timing of the GF unit. The other was the direction of impending collision, which was signaled by the relative location of the center of the activated LPLC2 population. If the center unit was located at the left side of the LPLC2 array, the robot should turn right to avoid the obstacle and vice versa.

Table 1. Parameter values for the model

| Parameter | Value |
|------------------------|-------------------|
| L_0 ^a | 2 |
| L_0 ^b | 1.5 |
| L_1 ^a | 2 |
| L_1 ^b | -2 |
| RF (in pixels) | 40 × 40~100 × 100 |
| w | 5~250 |
| τ (ms) | 30~300 |
| E_{leak} (mV) | -60 |
| V_{th} (mV) | -50 |
| V_{re} (mV) | -70 |

^aUsed for open-loop simulations except for those with real-world looming animals.

^bUsed for both closed-loop simulations on virtual robots and open-loop simulations with real-world looming animals.

We then checked the visual field scope of obstacle detection by using two robots. The simulation experiment was designed to make robot 2 (without the model) move straight toward stationary robot 1 (equipped with the model) at a constant speed. By testing various starting positions of robot 2, the simulation results found that robot 1 successfully avoided approaching robot 2 as long as robot 2 did not exceed the 120-degree field of view of robot 1 (Figure 7A).

We next examined the model's performance in a moving robot through a cluttered environment. One hundred cylindrical objects were randomly placed within a 50×50 m square arena as obstacles (Figure 7B). The robot started from a randomly given position and did not change the moving direction except when necessary to avoid an obstacle. Simulation results found that the robot performed collision avoidance very well in most cases, irrespective of the starting condition (Figure 7B), which was considered a high success rate (Table 2). The success rate is defined as $N_a/(N_a + N_m)$, where N_a is the collision avoidance time and N_m is the miss detection time.

Since the only purpose was to avoid collisions, the robot tended to stay in a relatively small area. To ensure that the robot travels through the environment, we set a goal location. Collision avoidance tasks during goal-directed traveling are more general in application. In each trial, the robot kept moving in a straight line toward the goal unless an obstacle was detected. After each collision avoidance action, the robot decided a goal-constrained movement direction. Simulation results found that the robot traveled through obstacles and reached the goal location with a high success rate, regardless of how these obstacles were distributed (Figures 7C and 7D) (Table 3). The success rate for the goal-directed tasks was defined as $N_s/(N_s + N_f)$, where N_s is the number of times the robot successfully reached the goal and N_f the number of times the robot failed to reach the goal.

Last, we examined the model's performance when facing obstacles more like real-world objects. Different from the black cylinders used above, four virtual obstacles mimicking real objects were disposed within a 10 × 10 m square arena (Figure 8A). A light above the arena center provided ample illumination. The scene contrast varied between the objects and even between different views of the same individual objects, causing the measurement efficiency of visual motion by the model to vary across the arena. Simulation results found that although a failure occurred in rare cases, the moving robot successfully avoided collisions most of the time (Figure 8B) (Table 2). The failure was mainly because the contrast of the front of an obstacle was too weak to be distinguished by the robot from that of its surrounding background (Figure 8C). In contrast, a successful collision avoidance was performed once the robot approached the rear of the same obstacle where the contrast was higher than that of its front surface (Figure 8D).

In summary, the tests show that the model is a robust and efficient algorithm for real-time collision avoidance that relies only on visual motion measured by an EMD array. Adjustable parameters of visual motion detection make the model adaptable to various robot speeds and body sizes.

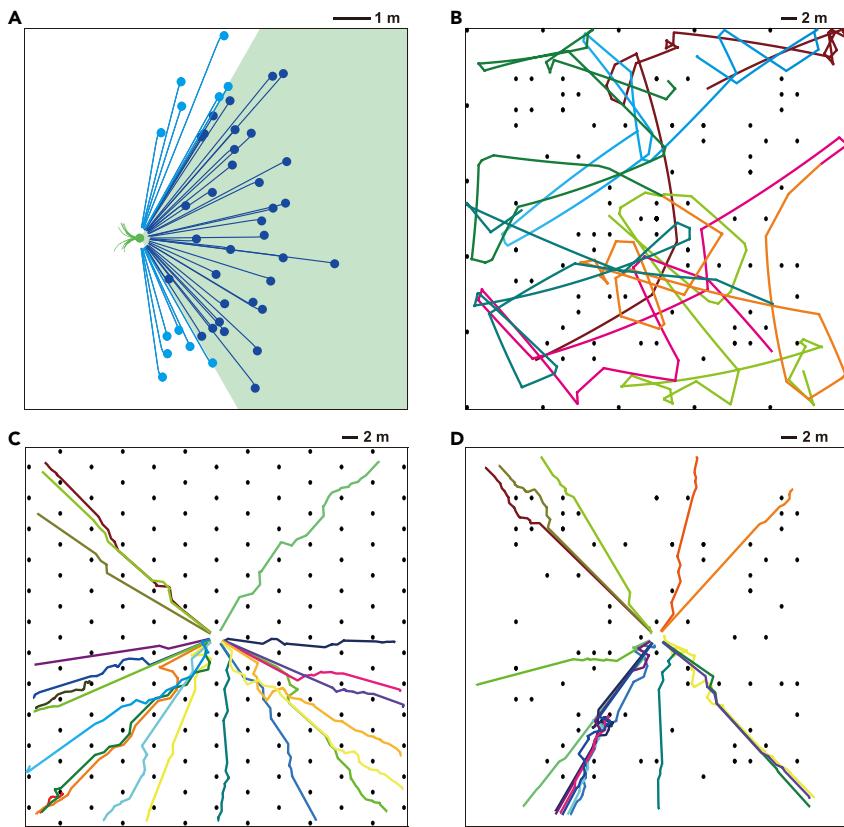


Figure 7. Obstacle detection and avoidance in cluttered environments

(A) Detection Scope. Robot 1 (green filled circle) was equipped with the model and kept stationary except for avoiding an obstacle (green curves: avoiding trajectories). Robot 2 without the model was moving toward robot 1 from various starting positions at a constant speed within a range of 0.1–0.9 m/s (light and dark blue filled circles and lines). Robot 1 successfully avoided robot 2 as long as robot 2 was within its 120-degree field of view (green area).

(B) Trajectories of a robot moving around obstacles. Ninety-two black cylinders with a diameter and height of 0.5 m were randomly arranged in the arena as obstacles with a minimum distance of 2 m (black spots). The robot has a size of $0.25 \times 0.25 \times 0.25$ m. The trajectories with different starting positions are color-coded.

(C and D) The same as (B) except that a goal location was set for the robot. The goal location is a 2×2 m region around the area center. Cylinder obstacles were either regularly arranged (C, 162 obstacles with a minimum distance of 4 m) or randomly placed (D, 72 obstacles with a minimum distance of 2 m).

Looming detection is robust to real-world objects

Looming objects used in the above open- and closed-loop simulations are not real-world objects. Real-world objects vary in features such as shape, texture, contrast, and luminance, just like those experienced by prey organisms. An obvious question to ask is whether the model works with real-world looming objects. To this end, we next tested the model on Videos taken in various real-world scenes, in each of which an animal was flying or running toward frame.

By presenting each of the Videos (see Videos S5, S6, S7, S8, S9, and S10) to the model, the simulations show that the looming animals were successfully detected (black curves in Figures 9B, 9C, 9D, 9E, 9F, and 9G) provided that the threshold parameter L_1 in Equation 5 was set as $L_1 < 0$ (see Table 1 for parameter values).

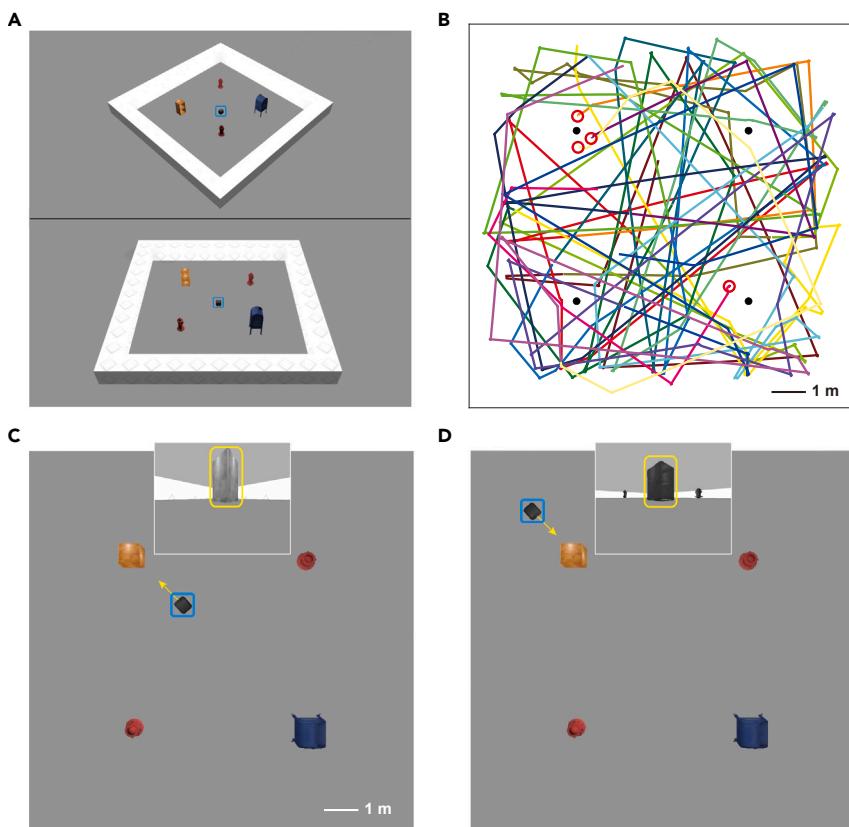
Table 2. Success rate of a moving robot without a goal direction

| Collision avoidance task | Avoidance times (N_a) | Miss times (N_m) | Success rate |
|--------------------------|---------------------------|----------------------|--------------|
| Figure 7B | 99 | 2 | 98.0% |
| Figure 8B | 41 | 4 | 91.1% |

Table 3. Success rate of a moving robot with a goal direction

| Collision avoidance task | Success times (N_s) | Failure times (N_f) | Success rate |
|--------------------------|-------------------------|-------------------------|--------------|
| Figure 7C | 20 | 2 | 90.9% |
| Figure 7D | 16 | 1 | 94.1% |

Our simulations under the condition of $L_1 > 0$ (see Table 1 for the values of parameters with superscript "a") show that the model was not able to guarantee a successful detection of looming animals, as shown by the red curves in Figures 9B, 9C, 9D, 9E, 9F, and 9G, indicating the significance of the setting of L_1 parameter. This is because the approaching animal was usually not flying or running straight toward camera. The animal thus induced visual motion patterns that expanded in three directions {1, 2, 3} but expanded more weakly or even contracted slightly in another direction {4}. For example, the cat during the time interval between two representative frames induced a visual motion pattern that slightly contracted in the leftward direction but expanded in the other three directions (left, Figure 9A; middle, Figure 9A). Similarly, another looming animal induced a visual motion pattern that expanded more weakly in the upward direction than in

**Figure 8. Obstacle detection and avoidance facing real-world objects**

(A) Scene arrangement in the test arena (10×10 m) taken from two different viewpoints. Around the robot (marked by the blue boundary), four objects were placed: two fire hydrants, a postbox, and a cabinet. A spotlight source is placed over the ground center.

(B) Trajectories of the robot moving through the arena. Different trials were color-coded, each of which lasted 5 min. The robot successfully avoided the obstacles (black spots) most of the time. A few collision events (red circles) were mainly caused by uneven illumination.

(C and D) Contrast effect on obstacle detection. The robot performed successful collision avoidance when approaching the rear of the golden cabinet (D), but it failed to avoid the same object when approaching from the opposite side (C). Compared with the former case (D, inset), the cabinet in the latter case (C, inset) was too weak to be discriminated from its background. One snapshot of the robot's field of view is displayed in the inset, in which the cabinet is marked by the yellow rectangle.

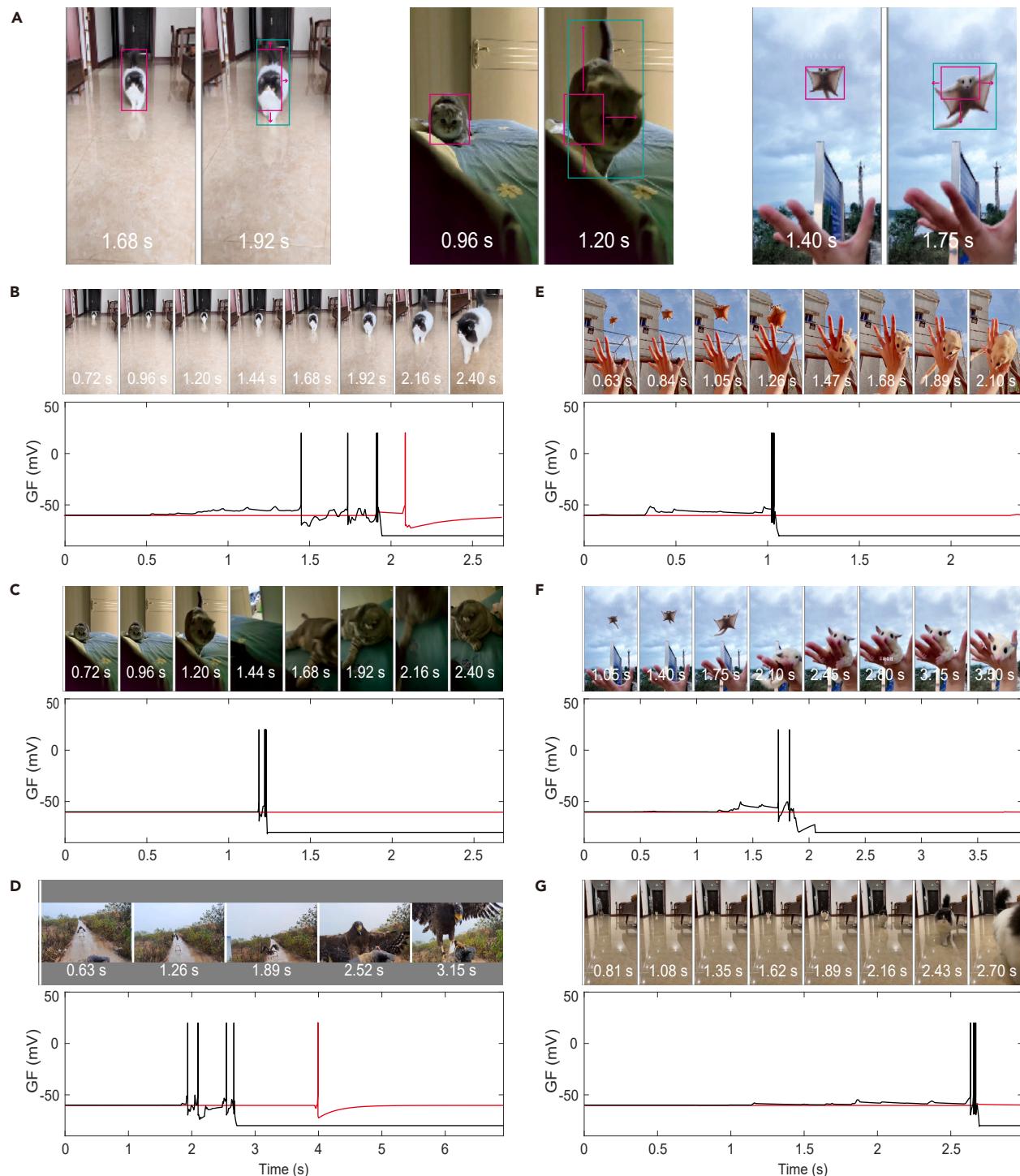


Figure 9. Looming detection is robust to real-world objects taken in various real-world scenes

(A) Schematic of a change in the area occupied by a looming animal in the visual field. Three pairs of frames on the left, middle, and right side were taken from Videos S5, S6, and S9, respectively. For each pair, the looming animal in the later frame (outlined by the green boundary) was nearer to the camera than in the earlier frame (outlined by the red boundary). During the time interval between the two frames, the animal induced a visual motion pattern that expanded strongly in three directions (three arrows) but either expanded more weakly (right) or contracted slightly (left; middle) in another direction.

Figure 9. Continued

(B) The simulated membrane potential of the GF unit on a direct collision course of the looming animal in [Video S5](#). Representative frames at the time stamp marked are displayed at the top. There are two steps of input preprocessing. First, the [Video frames in color were converted to grayscale](#). Second, the size of the grayscale images was reduced to 25 percent of the original images by [downsampling](#). The preprocessed frames were then fed to the EMD array consisting of $N \times M$ units, whose resolution matched the size of the preprocessed frames. Since the time step of the EMD array is 10 ms, the Video presented to the model was equivalently speeded up from 30 frames per second (fps) to 100 fps, as shown in the horizontal x axis.

(C, D, E, F, and G) Same as (B) except that the visual input to the model was replaced by [Videos S6, S7, S8, S9](#), and [S10](#), respectively.

the other three directions (right, [Figure 9A](#)). To ensure a successful detection of approaching animals with a small deviation angle relative to the camera, [the model must relax the detection range in one cardinal direction](#). Therefore, when facing these real-world looming objects, the model took a two-step detection in the same manner as described under closed-loop conditions. First, the LPLC2 layer kept detecting a flow pattern that expanded simultaneously [in any three cardinal directions](#). Once such a flow pattern was caught, the layer then checked whether the pattern satisfied the condition $[L_{ij}^4(t) - L_1]^+ > 0$ in another cardinal direction ([Equation 5](#)).



Our results with the setting of $L_1 < 0$ predicts that an individual LPLC2 neuron is able to be activated when the expandance strength of the visual motion pattern within its four-layered RF is not uniform across four cardinal directions, even if the pattern is slightly contracting in one of the four directions. To the best of our knowledge, existing experimental recordings on LPLC2 cells were not performed under the condition of stimuli characterizing the [real-world looming objects](#) (e.g., Klapoetke et al. ²⁰). Future studies, especially using electrophysiological approaches, are expected to test the prediction by designing such stimuli.

In summary, the model is robust and efficient for detecting looming animals in Videos taken in real-world scenes.

DISCUSSION

Testable predictions

Previous studies have shown that it is an [angular-size threshold reached by the looming stimulus](#) on the [retina](#) but not a fixed time before collision with the approaching object that determines the initiation of looming-mediated escape behaviors in animals including locusts,¹⁶ zebrafish⁵ and flies.^{19,22} Although types of [angular-size threshold detectors](#) (a neuron or population of neurons) have been identified, the circuit mechanism for the emergence of such an angular size threshold is still unclear. A prediction of the model is that the threshold angular-size is determined by the [RF size](#) of neurons upstream of the [angular-size threshold detectors](#). The prediction suggests that the angular-size threshold, which initiates the rapid escape behaviors in flies,^{19,22} should be determined by the RF size of LPLC2 neurons.

LPLC2 and LC4 visual projection neurons provide the GF descending neuron with two types of information about a looming object: its subtended [angular size](#) and its [angular velocity](#).^{19,22} How the information is specifically encoded still remains elusive. The model did not explicitly include LC4 units. Our simulations demonstrated that the looming velocity, inferred from the temporally dynamic change of the instantaneous number of [activated LPLC2 population](#) ([Equation 9](#) in [STAR Methods](#)), ensured the model a looming detection. We thus speculate that [LC4](#) neurons might extract the [looming velocity](#) from two input sources: the [motion pathway](#) via the direct postsynaptic projections from LPLC2 onto LC4 neurons¹⁹ and another unidentified non-LPLC2 pathway. The remaining [velocity tuning](#) of the GF neuron in [LPLC2-silenced flies](#)¹⁹ might originate from the [non-LPLC2 pathway](#), which might primarily depends on [luminance and contrast](#). Future investigations are expected to provide insights into the neural computation underlying the encoding of [looming velocity](#).

Comparison with previous bioinspired models

To solve collision avoidance, an elegant model sought to extract a relative nearness map from visual motion, i.e., the relative distance of an agent to objects in the surrounding environment.³⁶ The challenging extraction was achievable provided that the agent was confined to move in a plane and the eye of the agent was spherical. Our model seeks to extract expanding visual motion and is thus insensitive to the conditions. The nearness map must cover the visual field. Therefore, their model requires strongly textured images as inputs, and the FOE of the optic flow field must be dealt with separately as a singular point. By contrast, our

model readily detects a uniform looming object whose surface lacks optic flow, during which the FOE is naturally detected as the location of impending collision. Both models were established upon EMD-based visual motion measurement. The essential difference between two models can be attributed to the fact that our model is further inspired by the fly LPLC2–GF pathway that is downstream of the EMD circuit.

Another type of looming detection models is inspired by the lobula giant motion detectors (LGMDs) of locusts.^{15,18,37} The wide-field neurons LGMD1¹⁰ and LGMD2¹³ have no directionally selective inputs. Their motion sensitivity is derived from luminance temporal changes integrated by their giant dendritic arbor. This characteristic makes LGMD-inspired models respond not only to looming stimuli but also to receding and translating stimuli, which are different from our model that does not respond to non-looming stimuli. These models include various improvement mechanisms, such as parameter optimization, to increase their selectivity to looming stimuli.^{18,37}

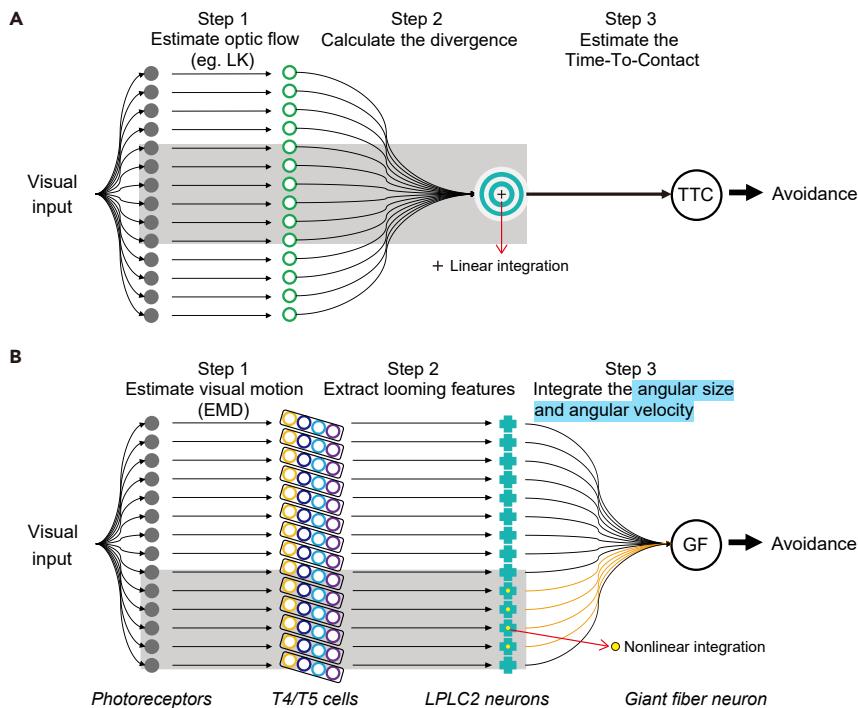
Different from most LGMD-inspired models, the connectionist model proposed by Bermúdez i Badia et al.¹⁵ can identify the direction of an impending collision at the LGMD's presynaptic layer, i.e., the second chiasma. Their model did not explicitly include any local motion detectors, but the pairwise combination of ON and OFF cells in the medulla layer actually detected visual motion. Leaving aside the different visual systems on which the two models are based, their model exhibits similar characteristics to ours from an computational point of view. Their second chiasma layer (i.e., the LGMD presynaptic fan-in) and the LGMD neuron unit computationally corresponds to our LPLC2 layer and GF unit, respectively. In addition, similar to the dependence of the angular-size threshold on the RF of the LPLC2 units in our model, the larger the RF of projection neurons in the second chiasma layer was, the larger the angular-size threshold was in their model. However, their model did not address how the individual projection neurons integrated the oriented contrast boundaries or directional visual motion in the four cardinal directions. Therefore, it is unclear whether the model proposed by Bermúdez i Badia et al.¹⁵ could completely suppress the responses to non-looming stimuli such as translating stimuli.

A recent study trained a two-layer network to detect impending collision objects.³⁸ Similar to our model, the first and second layer of their network mimicked an EMD array and a LPLC2 population, respectively. Unlike our model, the receptive field (i.e., its four spatial filters) of their individual LPLC2 units was not pre-defined but determined by a training procedure using artificial visual stimuli. Both their model and ours support that LPLC2 neurons employ a population strategy to ensure an efficient detection of impending collision. However, their model suggests that individual LPLC2 neurons additively integrate visual motion flow expanding in the four cardinal directions, whereas ours argues for multiplicative integration. Future studies, especially using electrophysiological approaches, are expected to provide new insight into the integration property across LPLC2 dendrites.

Comparison with optic flow-based algorithms

Optic flow-based algorithms for collision detection feature lightweight and low-cost sensors such as single cameras, making them advantageous in applications where constraints in weight and computational power are important.³⁹ A typical algorithm measures flow field divergence at the image center (assumed to be the FOE) and then acquires an estimation of the time-to-contact (TTC).^{40–42} TTC is the time it would take an agent to make contact with a surrounding obstacle if the agent's velocity is maintained constant. Positive flow divergence indicates the presence of potential obstacles with a TTC inversely proportional to the divergence magnitude.

Measuring flow divergence is equivalent to extracting expansive visual motion in our model, but the two algorithms are essentially different in the implementation method (Figure 10). The flow divergence algorithm calculates the sum of partial spatial derivatives of optic flow components in orthogonal directions, whereas our model argues for a multiplication integration of visual motion across the four cardinal directions. Moreover, the flow divergence algorithm relies entirely on the FOE's divergence, whereas our model lets LPLC2 neurons around the FOE make a collective voting. The RF of LPLC2 neurons naturally determines an RF-dependent peak firing in the GF unit for signaling the anticipated timing of impending collisions in our model, whereas the flow divergence is continuously increasing on the collision course of the looming object. Finally, the flow divergence algorithm measures only instantaneous divergence values, whereas our model utilizes the GF unit to accumulate the looming confidence over time up to a firing threshold.

**Figure 10. Comparison between the flow divergence algorithm and the fly inspired model**

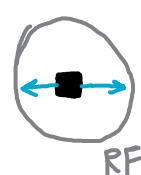
Schematic of the flow divergence algorithm (A) and the fly inspired Model (B) for collision detection. Their differences are reflected in the 3 steps. Step 1: Gradient-based methods for measuring optic flow (A) vs. the EMD method for analyzing visual motion (B). Step 2: a summation of partial spatial derivatives of optic flow components in orthogonal directions (A, "+" sign) vs. a multiplication integration of visual motion across the four cardinal directions (B, "o" signs); relying on the FOE's flow divergence alone (A, concentric circles) vs. adopting a distributed coding strategy (B, green cross symbols). Step 3: Transforming an instantaneous measurement of flow divergence into a TTC value (A) vs. accumulating the looming confidence over time up to a firing threshold by the GF unit (B). The gray shadow in (A) and (B) represents the pathway activated by a looming stimulus.

The flow divergence algorithm relies considerably on rich texture information and is susceptible to optic flow noise. To make an accurate TTC estimation, the flow divergence algorithm was improved by introducing either an extra stabilization mechanism to keep the FOE aligned with the camera's optic axis⁴¹ or an extra temporal module to integrate the confidence of a TTC estimation over time.⁴⁰ This is in contrast to our model, which is robust even when facing textureless looming obstacles. On the other hand, the shortcoming of our model is its inability to detect a textureless looming object once its angular size on the modeled eye exceeds the RF of LPLC2 units. Therefore, our model is more applicable for detecting objects approaching from a distance, even if the objects are textureless.

Correlation-type EMDs are generally adopted by biological vision for visual motion measurement and have a much lower computational complexity than computer vision methods such as gradient-based algorithms.²³ However, EMDs are seldom used in real-time applications because of their well-known sensitivity to nonmotion-related factors such as luminance contrast and texture properties. Our model proposes a fly inspired solution to cope with EMD sensitivity in looming detection, thus shedding light on real-time applications of EMDs.

Limitations of the study

As a first limitation, the model cannot quantitatively reproduce the physiological response properties of individual LPLC2 neurons under some specially laboratory-designed stimuli. For instance, one-dimensional narrow bar expanding toward two but not four cardinal directions (at the top of Figure 3E in Klapoetke et al.²⁰) is a laboratory-designed stimulus, which is different from the real profile of an object's subtended angle during looming in the physical world. Under this stimulus condition, the response strength of our



LPLC2 model units will depend on the bar expansion direction. By contrast, the observed LPLC2's responses were nearly uniform across all the expansion directions (at the top of Figure 3E in Klapoetke et al.²⁰). In another instance, the response strength of our LPLC2 model units will not be strongly affected by tangential motion at the periphery of the LPLC2's RF. On the contrary, the observed LPLC2's responses were significantly inhibited by the presence of tangential motion at RF edges (Figures 4G, 4C and 4D in Klapoetke et al.²⁰). The discrepancy between the model and experimental recordings at the level of individual LPLC2 neurons suggests that for each of the four sub-RFs, visual motion inducing directionally selective inhibition to the LPLC2 is not restricted to the null direction of the corresponding sub-RF. Other visual motion orthogonal to the sub-RF's null direction may induce inhibitory input to the LPLC2, too. A surrounding inhibitory region at the boundaries of the four-layered RF of individual LPLC2 units, which was not considered in our model, should play a role in the observed LPLC2's responses to the specially designed stimuli. Predefining the detailed structure or the fine spatiotemporal characteristics of the hypothesized RF, however, is a daunting task. Training the model network using the specially designed stimuli, in a manner similar to that adopted by Zhou et al.,³⁸ could be helpful for determining the hypothesized RF. Future investigations are expected to provide insights into the additional mechanisms involved in the visual feature processing of LPLC2.

→ LPi
Nature 2017
P5

As a second limitation, we used a planar screen as the modeled eye for simplicity (Figure 2C), whose field of view was smaller than that of the hemispherical compound eyes of the fly. The planar screen introduced spatial distortions in the peripheral field of view. This can be illustrated by the example of distorted cylindrical objects shown in Figure 6A4, which were seen by the planar camera of the virtual robot. The distortions could make the model have different optimal parameter values between the central and peripheral field of view. Therefore, it is the spherical but not planar array of visual sensors that can take full advantage of the model algorithm.

Finally, we did not test the model under high-speed collision conditions due to the limitations of our existing software and hardware resources. The current study mainly focuses on establishing the fly inspired algorithm and demonstrating its effectiveness and potential advantage in real-time scenarios. Future studies should exploit the speed advantage of the algorithm in real walking or flying robots by optimizing the algorithm and using airborne computer programs.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
 - Lead contact
 - Materials availability
 - Data and code availability
- METHOD DETAILS
 - Visual stimuli
 - EMD array
 - LPLC2 layer
 - GF unit
 - Model simulations

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at <https://doi.org/10.1016/j.isci.2023.106337>.

ACKNOWLEDGMENTS

We would like to thank Qiyue Jiang, Xiaodan Zhang, Daniel Ocean, and Meikun for kindly providing Videos taken by themselves. This work was supported by the National Natural Science Foundation of China (Grants 31871050 and 31271172 to Z.W., 32061143011 and 31970947 to Y.L.), the Strategic Priority Research Program of the Chinese Academy of Sciences (XDB32000000 to A.G.), and National Key Research and Development Program of China (2019YFA0802402 to A.G. and Y.L.).

AUTHOR CONTRIBUTIONS

J.Z., Z.W., and A.G. conceptualized the study. J.Z. and Z.W. designed the simulation experiments. S.X. and J.Z. wrote the programs of visual stimuli. J.Z. performed the simulations. S.X., A.G., and Y.L. analyzed the implications and contributed ideas. J.Z. and Z.W. wrote the original draft. J.Z., Z.W., A.G., and Y.L. edited versions of the article.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: September 16, 2022

Revised: December 26, 2022

Accepted: March 1, 2023

Published: March 5, 2023

REFERENCES

1. Falanga, D., Kleber, K., and Scaramuzza, D. (2020). Dynamic obstacle avoidance for quadrotors with event cameras. *Sci. Robot.* 5, eaaz9712. <https://doi.org/10.1126/scirobotics.aaz9712>.
2. Card, G., and Dickinson, M. (2008). Performance trade-offs in the flight initiation of *Drosophila*. *J. Exp. Biol.* 211, 341–353. <https://doi.org/10.1242/jeb.012682>.
3. Oliva, D., Medan, V., and Tomsic, D. (2007). Escape behavior and neuronal responses to looming stimuli in the crab *Chasmagnathus granulatus* (Decapoda: Grapsidae). *J. Exp. Biol.* 210, 865–880. <https://doi.org/10.1242/jeb.02707>.
4. Schiff, W., Caviness, J.A., and Gibson, J.J. (1962). Persistent fear responses in rhesus monkeys in response to the optical stimulus of "looming". *Science* 136, 982–983. <https://doi.org/10.1126/science.136.3520.982>.
5. Temizer, I., Donovan, J.C., Baier, H., and Semmelhack, J.L. (2015). A visual pathway for looming-evoked escape in larval zebrafish. *Curr. Biol.* 25, 1823–1834. <https://doi.org/10.1016/j.cub.2015.06.002>.
6. Yilmaz, M., and Meister, M. (2013). Rapid innate defensive responses of mice to looming visual stimuli. *Curr. Biol.* 23, 2011–2015. <https://doi.org/10.1016/j.cub.2013.08.015>.
7. de Vries, S.E.J., and Clandinin, T.R. (2012). Loom-sensitive neurons link computation to action in the *Drosophila* visual system. *Curr. Biol.* 22, 353–362. <https://doi.org/10.1016/j.cub.2012.01.007>.
8. Dunn, T.W., Gebhardt, C., Naumann, E.A., Rieger, C., Ahrens, M.B., Engert, F., and Del Bene, F. (2016). Neural circuits underlying visually evoked escapes in larval zebrafish. *Neuron* 89, 613–628. <https://doi.org/10.1016/j.neuron.2015.12.021>.
9. Liu, Y.J., Wang, Q., and Li, B. (2011). Neuronal responses to looming objects in the superior colliculus of the cat. *Brain Behav. Evol.* 77, 193–205. <https://doi.org/10.1159/000327045>.
10. O'Shea, M., and Williams, J.L.D. (1974). The anatomy and output connection of a locust visual interneurone; the lobular giant movement detector (LGMD) neurone. *J. Comp. Physiol.* 91, 257–266.
11. Peek, M.Y., and Card, G.M. (2016). Comparative approaches to escape. *Curr. Opin. Neurobiol.* 41, 167–173. <https://doi.org/10.1016/j.conb.2016.09.012>.
12. Shang, C., Chen, Z., Liu, A., Li, Y., Zhang, J., Qu, B., Yan, F., Zhang, Y., Liu, W., Liu, Z., et al. (2018). Divergent midbrain circuits orchestrate escape and freezing responses to looming stimuli in mice. *Nat. Commun.* 9, 1232. <https://doi.org/10.1038/s41467-018-03580-7>.
13. Simmons, P.J., and Rind, F.C. (1997). Responses to object approach by a wide field visual neurone, the LGMD2 of the locust: characterization and image cues. *J. Comp. Physiol. Sens. Neural Behav. Physiol.* 180, 203–214. <https://doi.org/10.1007/s003590050041>.
14. Fraser Rowell, C.H., O'Shea, M., and Williams, J.L. (1977). The neuronal basis of a sensory analyser, the acridid movement detector system. IV. The preference for small field stimuli. *J. Exp. Biol.* 68, 157–185. <https://doi.org/10.1242/jeb.68.1.157>.
15. Bermúdez i Badia, S., Bernardet, U., and Verschure, P.F.M.J. (2010). Non-linear neuronal responses as an emergent property of afferent networks: a case study of the locust lobula giant movement detector. *PLoS Comput. Biol.* 6, e1000701. <https://doi.org/10.1371/journal.pcbi.1000701>.
16. Gabbiani, F., Krapp, H.G., and Laurent, G. (1999). Computation of object approach by a wide-field, motion-sensitive neuron. *J. Neurosci.* 19, 1122–1141. <https://doi.org/10.1523/JNEUROSCI.19-03-01122.1999>.
17. Gabbiani, F., Krapp, H.G., Koch, C., and Laurent, G. (2002). Multiplicative computation in a visual neuron sensitive to looming. *Nature* 420, 320–324. <https://doi.org/10.1038/nature01190>.
18. Rind, F.C., and Bramwell, D.I. (1996). Neural network based on the input organization of an identified neuron signaling impending collision. *J. Neurophysiol.* 75, 967–985. <https://doi.org/10.1152/jn.1996.75.3.967>.
19. Ache, J.M., Polksy, J., Alghailani, S., Parekh, R., Breads, P., Peek, M.Y., Bock, D.D., von Reyn, C.R., and Card, G.M. (2019). Neural basis for looming size and velocity encoding in the *Drosophila* giant fiber escape pathway. *Curr. Biol.* 29, 1073–1081.e4. <https://doi.org/10.1016/j.cub.2019.01.079>.
20. Klapoetke, N.C., Nern, A., Peek, M.Y., Rogers, E.M., Breads, P., Rubin, G.M., Reiser, M.B., and Card, G.M. (2017). Ultra-selective looming detection from radial motion opponency. *Nature* 551, 237–241. <https://doi.org/10.1038/nature24626>.
21. von Reyn, C.R., Breads, P., Peek, M.Y., Zheng, G.Z., Williamson, W.R., Yee, A.L., Leonardo, A., and Card, G.M. (2014). A spike-timing mechanism for action selection. *Nat. Neurosci.* 17, 962–970. <https://doi.org/10.1038/nn.3741>.
22. von Reyn, C.R., Nern, A., Williamson, W.R., Breads, P., Wu, M., Namiki, S., and Card, G.M. (2017). Feature integration drives probabilistic behavior in the *Drosophila* escape response. *Neuron* 94, 1190–1204.e6. <https://doi.org/10.1016/j.neuron.2017.05.036>.
23. Barron, J.L., Fleet, D.J., and Beauchemin, S.S. (1994). Performance of optical flow techniques. *Int. J. Comput. Vis.* 12, 43–77. <https://doi.org/10.1007/BF01420984>.
24. Borst, A., and Helmstaedter, M. (2015). Common circuit design in fly and mammalian motion vision. *Nat. Neurosci.* 18, 1067–1076. <https://doi.org/10.1038/nn.4050>.
25. Clifford, C.W.G., and Ibbotson, M.R. (2002). Fundamental mechanisms of visual motion detection: models, cells and functions. *Prog. Neurobiol.* 68, 409–437. [https://doi.org/10.1016/s0301-0082\(02\)00154-5](https://doi.org/10.1016/s0301-0082(02)00154-5).
26. Wolf-Oberholzer, F., and Kirschfeld, K. (1994). Motion sensitivity in the nucleus of the basal optic root of the pigeon. *J. Neurophysiol.* 71, 1559–1573. <https://doi.org/10.1152/jn.1994.71.4.1559>.

27. Hassenstein, B., and Reichardt, W. (1956). System theoretical analysis of time, sequence and sign analysis of the motion perception of the snout-beetle *Chlorophanus*. *Zeitschrift für Naturforschung B* 11, 513–524.
28. Buchner, E. (1984). Behavioural analysis of spatial vision in insects. In *Photoreception and vision in invertebrates*, M.A. Ali, ed. (Plenum), pp. 561–621.
29. Reichardt, W. (1987). Evaluation of optical motion information by movement detectors. *J. Comp. Physiol.* 161, 533–547. <https://doi.org/10.1007/BF00603660>.
30. Pouget, A., Dayan, P., and Zemel, R. (2000). Information processing with population codes. *Nat. Rev. Neurosci.* 1, 125–132. <https://doi.org/10.1038/35039062>.
31. Eichner, H., Joesch, M., Schnell, B., Reiff, D.F., and Borst, A. (2011). Internal structure of the fly elementary motion detector. *Neuron* 70, 1155–1164. <https://doi.org/10.1016/j.neuron.2011.03.028>.
32. Borst, A., Haag, J., and Mauss, A.S. (2020). How fly neurons compute the direction of visual motion. *J. Comp. Physiol. A Neuroethol. Sens. Neural Behav. Physiol.* 206, 109–124. <https://doi.org/10.1007/s00359-019-01375-9>.
33. Haag, J., Arenz, A., Serbe, E., Gabbiani, F., and Borst, A. (2016). Complementary mechanisms create direction selectivity in the fly. *eLife* 5, e17421. <https://doi.org/10.7554/eLife.17421>.
34. Groschner, L.N., Malis, J.G., Zuidinga, B., and Borst, A. (2022). A biophysical account of multiplication by a single neuron. *Nature* 603, 119–123. <https://doi.org/10.1038/s41586-022-04428-3>.
35. YouTube. (2019). Driving in Switzerland 6: from grindelwald to lauterbrunnen [Video file]. https://www.youtube.com/watch?v=b-WViLMs_4c&t=573s.
36. Bertrand, O.J.N., Lindemann, J.P., and Egelhaaf, M. (2015). A bio-inspired collision avoidance model based on spatial information derived from motion detectors leads to common routes. *PLoS Comput. Biol.* 11, e1004339. <https://doi.org/10.1371/journal.pcbi.1004339>.
37. Fu, Q., Hu, C., Peng, J., Rind, F.C., and Yue, S. (2020). A robust collision perception visual neural network with specific selectivity to darker objects. *IEEE Trans. Cybern.* 50, 5074–5088. <https://doi.org/10.1109/TCYB.2019.2946090>.
38. Zhou, B., Li, Z., Kim, S., Lafferty, J., and Clark, D.A. (2022). Shallow neural networks trained to detect collisions recover features of visual loom-selective neurons. *eLife* 11, e72067. <https://doi.org/10.7554/eLife.72067>.
39. Hoy, M., Matveev, A.S., and Savkin, A.V. (2015). Algorithms for collision-free navigation of mobile robots in complex cluttered environments: a survey. *Robotica* 33, 463–497. <https://doi.org/10.1017/S0263574714000289>.
40. Coombs, D., Herman, M., Hong, T., and Nashman, M. (1998). Real-time obstacle avoidance using central flow divergence, and peripheral flow. *IEEE Trans. Rob. Autom.* 14, 49–59. <https://doi.org/10.1109/70.660840>.
41. McCarthy, C., Barnes, N., and Mahony, R. (2008). A robust docking strategy for a mobile robot using flow field divergence. *IEEE Trans. Robot.* 24, 832–842. <https://doi.org/10.1109/TRO.2008.926871>.
42. Nelson, R.C., and Aloimonos, J. (1989). Obstacle avoidance using flow field divergence. *IEEE Trans. Pattern Anal. Mach. Intell.* 11, 1102–1106. <https://doi.org/10.1109/34.42840>.
43. Mauss, A.S., Pankova, K., Arenz, A., Nern, A., Rubin, G.M., and Borst, A. (2015). Neural circuit to integrate opposing motions in the visual field. *Cell* 162, 351–362. <https://doi.org/10.1016/j.cell.2015.06.035>.

STAR★METHODS

KEY RESOURCES TABLE

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|------------------------------|---------------|---|
| Software and algorithms | | |
| MATLAB 2019b | MathWorks | RRID: SCR_001622 |
| Ubuntu 16.04 LTS | Canonical Ltd | https://ubuntu.com/ |
| Robot Operating System (ROS) | Open Robotics | https://www.ros.org/ |
| Gazebo | Open Robotics | https://www.ros.org/ |
| Code | This paper | https://github.com/zhaojunyu01/EMD-LPLC2-GF-model.git |
| Deposited data | | |
| Data | This paper | https://github.com/zhaojunyu01/EMD-LPLC2-GF-model.git |

RESOURCE AVAILABILITY

Lead contact

Requests for further information regarding resources used in this study should be directed to the lead contact, Zhihua Wu (wuzh@shu.edu.cn).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- Data necessary to reproduce the figures have been deposited at the GitHub repository: <https://github.com/zhaojunyu01/EMD-LPLC2-GF-model.git> and are publicly available as of the date of publication. The URL is listed in the [key resources table](#).
- Code necessary to reproduce the figures has been deposited at the GitHub repository: <https://github.com/zhaojunyu01/EMD-LPLC2-GF-model.git> and is publicly available as of the date of publication. The URL is listed in the [key resources table](#).
- Any additional information required to reanalyze the data reported in this paper is available from the [lead contact](#) upon request.

METHOD DETAILS

Visual stimuli

Four types of stimuli were created for simulations. The resolution of each frame image is denoted as $N \times M$ pixels. Input images have the same size as the entire frame. The first type shares a static background and were used for simulations, the results of which are shown in [Figures 2, 3, and 4](#). The input images were pre-processed to $N \times M = 200 \times 150$ pixels (i.e., $118^\circ \times 103^\circ$) by downsampling. Except for the stimuli used in the 2nd–10th column of [Figures 2A](#) and [2B](#) that had a constant edge speed, all the other stimuli in [Figures 2, 3](#) and [4](#) mimicked a looming square object approaching the modeled eye with a constant velocity. We use the parameter L/v size-to-velocity ratio to characterize the looming object's approaching kinematics.¹⁶ The object's half-size is L , and its approaching velocity is v ([Figure 2C](#)). In the 2nd column of [Figures 2A](#) and [2B](#), the square expands from 6×6 pixels to 105×105 pixels, and all the cross arms of the stimuli in [Figures 2A](#) and [2B](#) have a width of 30 pixels.

The second type of stimuli was created by superimposing a looming square upon a scene captured by a driving recorder mounted on a vehicle moving forward³⁵ or a translational natural scene. The input images were preprocessed to $N \times M = 400 \times 300$ pixels by downsampling and used for simulations, the results of which are shown in [Figure 5](#). The third type of stimuli was a series of images captured by the camera of

a virtual robot while moving within its surrounding environment. The input images were preprocessed to $N \times M = 300 \times 250$ pixels by downsampling and used for simulations, the results of which are shown in Figures 6, 7, and 8. The fourth type of stimuli was Videos S5, S6, S7, S8, S9, and S10. They were used as the visual input to the model for simulations, the results of which are shown in Figure 9. The six Videos were taken in various indoor and outdoor scenes.

EMD array

The EMD array was developed with each unit receiving two inputs separated in visual space. The array consists of $N \times M$ units, whose resolution matches the size of the input images. Individual EMD units comprise parallel ON and OFF detector subunits (Figure 1B), whose detailed structure and parameter values are in accordance with a previous work by Eichner et al.³¹ (see Figure 4A in Eichner et al.³¹). Significant progress has been made in models of the fly EMD over the past decade.^{32,34} Our EMD model does not include the DC components of the Eichner model,³¹ which has no longer been justified by the progress. Visual inputs were first processed by a first-order high-pass filter with time constant $\tau_{HP} = 250$ ms. The filtered components were fed into two parallel half-wave rectifiers, one forming the ON pathway and the other the OFF pathway. The cutoff for ON and OFF rectifiers was set at zero and 0.05, respectively. First-order low-pass filters with $\tau_{LP} = 50$ ms were used in both ON and OFF subunits. The two mirror-symmetrical half-detectors of each horizontal full EMD units were modeled to detect rightward and leftward motion, respectively. The two mirror-symmetrical half-detectors of each vertical full EMD units were modeled to detect downward and upward motion, respectively.

To further simplify the problem, outputs of the two EMD arrays in the ON and OFF pathways were retinotopically added to give the final output of the ON+OFF EMD array. The rightward and leftward motion components detected by the ON+OFF EMD array are denoted as matrices $H_{ij}^r(t)$ and $H_{ij}^l(t)$, respectively. The downward and upward motion components detected by the ON+OFF EMD array are denoted as matrices $V_{ij}^d(t)$ and $V_{ij}^u(t)$, respectively. The four visual motion patterns were then projected to the LPLC2 layer (Figure 1), mimicking the four-layered projection from T4/T5 cells to the four distinct fields formed by dendritic arbors of individual LPLC2 neurons in the lobula plate.²⁰

LPLC2 layer

The LPLC2 layer consists of $N \times M$ units. Each LPLC2 unit integrates inputs from the EMD array in its receptive field (RF) with a four-layered structure denoted as K_r, K_l, K_d , and K_u (Figure 1C). The subscripts {r, l, d, u} denote rightward, leftward, downward, and upward, respectively. The RF is a cross in a square, and the arm width of the cross is one-third the length of the side of the square. The square was set as 100×100 pixels, i.e., $60^\circ \times 80^\circ$, unless otherwise specified. The adjacent units have highly overlapping RFs. Four input matrices to the LPLC2 layer were specifically calculated as $\{H_{ij}^r(t) - H_{ij}^l(t)\}$, $\{H_{ij}^l(t) - H_{ij}^r(t)\}$, $\{V_{ij}^d(t) - V_{ij}^u(t)\}$, and $\{V_{ij}^u(t) - V_{ij}^d(t)\}$ to simulate motion opponency induced by inhibitory inputs of lobula plate LPi interneurons in *Drosophila*.^{20,43} By convolving each of the four matrices with an RF kernel in the corresponding directions, we obtained inputs to LPLC2 units denoted as $L_{ij}^r(t)$, $L_{ij}^l(t)$, $L_{ij}^d(t)$, and $L_{ij}^u(t)$.

$$L_{ij}^r(t) = \{H_{ij}^r(t) - H_{ij}^l(t)\} * K_r \quad (\text{Equation 1})$$

$$L_{ij}^l(t) = \{H_{ij}^l(t) - H_{ij}^r(t)\} * K_l \quad (\text{Equation 2})$$

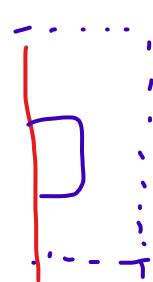
$$L_{ij}^d(t) = \{V_{ij}^d(t) - V_{ij}^u(t)\} * K_d \quad (\text{Equation 3})$$

$$L_{ij}^u(t) = \{V_{ij}^u(t) - V_{ij}^d(t)\} * K_u \quad (\text{Equation 4})$$

The state of individual LPLC2 units was simulated as a function of their total inputs, which are hypothesized to be a multiplicative integration of visual motion expanding in the four cardinal directions in their RFs.

$$LPLC2_{ij}(t) = [L_{ij}^1(t) - L_0]^+ \cdot [L_{ij}^2(t) - L_0]^+ \cdot [L_{ij}^3(t) - L_0]^+ \cdot [L_{ij}^4(t) - L_1]^+ \quad (\text{Equation 5})$$

where the function $[x]^+$ is defined as $\max(x, 0)$, and L_0 and L_1 are input thresholds with $L_0 > 0$. The superscripts {1, 2, 3, 4} in Equation 5 indicate {r, l, d, u} via one-to-one correspondence.



For open-loop simulations in Figures 2, 3, 4, and 5, we set $L_1 = L_0$. Therefore, the four input channels were equivalent, causing an LPLC2 unit to be activated only when it detects visual motion simultaneously expanding in all four cardinal directions. For open-loop simulations with real-world animals in Figure 9, we set $L_1 < 0$. This is because visual motion patterns, induced by the looming animals, usually simultaneously expand in three directions {1, 2, 3} but either expand more weakly or contract slightly in another direction {4}. Once the model was operating in the modality of closed-loop simulations (Figures 6, 7, and 8), we also set $L_1 < 0$ by considering that the robot is not an ideal mass point but has a definite body size (Figure 6C). The negative L_1 makes an LPLC2 unit activated when it detects visual motion patterns simultaneously expanding in three directions {1, 2, 3} and either expanding more weakly or contracting slightly in another direction {4}.

GF unit

The single GF neuron unit comprises the last layer of the model (Figure 1A). We used the leaky integrate-and-fire model to simulate the unit without loss of generality. By integrating inputs from the LPLC2 layer, the simulated membrane potential $V(t)$ of the unit obeyed the following dynamics.

$$\tau_m \frac{dV(t)}{dt} = -V(t) + E_{\text{Leak}} + R I^{\text{syn}}(t) \quad (\text{Equation 6})$$

in which an action potential is generated whenever $V(t)$ reaches a threshold value V_{th} . After an action potential, the potential is reset to a value V_{re} with $V_{\text{re}} < V_{\text{th}}$. Parameter R was set to be one as the simulated membrane resistance, and τ_m is the simulated membrane time constant. $I^{\text{syn}}(t)$ is the input current that is hypothesized to be proportional to a multiplicative integration of two quantities extracted from the LPLC2 layer: the number $N_{\text{act}}(t)$ of instantaneously activated LPLC2 units and its changing speed $\Delta N_{\text{act}}(t)/\Delta t$.

$$I^{\text{syn}}(t) = w \cdot N_{\text{act}}(t) \cdot \frac{\Delta N_{\text{act}}(t)}{\Delta t} \quad (\text{Equation 7})$$

$$N_{\text{act}}(t) = \sum_{i,j} S_{i,j}(t) \quad (\text{Equation 8})$$

$$\frac{\Delta N_{\text{act}}(t)}{\Delta t} = (N_{\text{act}}(t) - N_{\text{act}}(t - 1)) / \text{step} \quad (\text{Equation 9})$$

where step = 10 ms is the time interval between frames in visual stimulus inputs. Parameter w was used as a coefficient for scaling the input current $I^{\text{syn}}(t)$ to a reasonable range. To prevent a meaningless decrease in the simulated GF membrane potential induced by a negative changing speed $\Delta N_{\text{act}}(t)/\Delta t$ after the peak time of $N_{\text{act}}(t)$, a minimum potential of -80 mV was set for the model $V(t)$. $S_{i,j}(t)$ is the output of the LPLC2 layer, which was simulated to be a binary value matrix depending on the state of LPLC2 units in a nonlinear way.

$$S_{i,j}(t) = \begin{cases} 1, & \text{LPLC2}_{i,j}(t) > 0 \\ 0, & \text{LPLC2}_{i,j}(t) \leq 0 \end{cases} \quad (\text{Equation 10})$$

Model simulations

Simulation programs were run on a Dell Precision T7920 workstation under Windows 10. The EMD array and LPLC2 layer were simulated as a discrete-time system with a step of 10 ms, while the GF unit was a continuous-time system that was integrated using the fourth-order Runge–Kutta method with a time step of 0.5 ms. To link up different time scales, the EMD array and LPLC2 layer were updated every 10 ms, during which the GF unit was successively integrated for 20 steps. All the parameters are summarized in Table 1.

The open-loop simulations were coded in MATLAB 2019b (MathWorks). The closed-loop simulations were performed by communications between the EMD–LPLC2–GF model under Windows and virtual robots under Ubuntu 16.04. Specifically, we installed Ubuntu 16.04 into Windows via VMware Workstation Pro, into which the Robot Operating System (ROS) and physical simulation environment Gazebo were then installed. By utilizing the ROS interface of the MATLAB ROS toolbox, the EMD–LPLC2–GF model programmed in MATLAB was able to collect real-time visual input signals from a camera mounted on ROS controlled TurtleBot 3.0 virtual robots (specific type: turtlebot3_waffle_pi). The robot camera has a 120-degree field of view.