

# 广州大学

## 本科毕业设计

课题名称	注意力引导的 LPLC2 视觉碰撞感知模型
学 院	数学与信息科学学院
专 业	信息与计算科学
班级名称	信计 202
学生姓名	吴梓翀
学 号	2015200069
指导教师	傅沁冰
完成日期	2024-4-18

教 务 处 制

# 注意力引导的 LPLC2 视觉碰撞感知模型

信计 202 吴梓翀

指导教师:傅沁冰

**摘要** 视觉碰撞检测的关键在于及时感知物体在纵深方向上的靠近运动。而碰撞感知的选择性更为重要,即只对物体的接近做出强烈反应,而对其他运动(如远离或平移)保持沉默。近年来,视觉碰撞检测技术被广泛应用于无人机和机器人等智能机器中。由于生物的视觉系统经过数百万年的进化,能够高效、稳健地检测碰撞运动,并具有超强的碰撞选择性。因此一些受果蝇或蝗虫视觉神经启发的碰撞感知模型被提出并成功应用于检测不同场景下的碰撞运动。近年来,生物学家在果蝇的视觉神经中发现了一组具有极端碰撞选择性的神经元——小叶板/小叶柱 II (LPLC2) 神经簇:它的极端性在于只对从感受视野中心位置并且径向运动的物体做出反应。那么在此基础上,如何实现对从其他视野象限的径向运动的物体做出反应是目前算法研究中的空缺。因此,针对来自各个视野象限的物体碰撞的检测问题,本文聚焦与果蝇大脑内的 T4/T5 细胞产生的运动定位信息展开研究和建立模型。本文提出的由注意力机制引导的 LPLC2 碰撞感知模型,将接收到的运动定位信息进行处理,重新定位视野中心,并以新的视野中心进行计算物体沿着水平和垂直的四个基本方向的运动,随后将计算的结果映射到的四个象限相乘输出,从而实现 LPLC2 模型对其他象限碰撞感知。

为充分验证所提出的模型,本文实验构建了一系列的视频——包括在简单和复杂背景下,来自各个象限的迫近运动场景与多个象限不同对比度的物体的迫近运动场景,并于经典的 LGMD 神经网络模型进行对比实验。上述实验的结果表明,注意力引导的 LPLC2 模型能够有效的检测出来自各个象限的迫近运动物体,特别的是在不同象限同时有多个不同对比度的迫近物体时,该模型会更加倾向于识别高对比度的迫近物体。引入注意力引导机制能够很大程度上提高 LPLC2 模型的合理性与鲁棒性。

**关键词** 果蝇视觉系统; LPLC2; 神经网络模型; 极端碰撞选择性; 注意力机制

**ABSTRACT** The key to visual collision detection is the timely perception of approaching movements of objects in the longitudinal direction. And the selectivity of collision perception is even more important, i.e., it only reacts strongly to the approach of an object and remains silent to other motions (e.g., moving away or translating). In recent years, visual collision detection techniques have been widely used in intelligent machines such as drones and robots. Since the visual system of living beings has evolved over millions of years, it is capable of detecting collision motions efficiently and robustly with superb collision selectivity. Therefore some collision perception models inspired by the visual nerves of fruit flies or locusts have been proposed and successfully applied to detect collision motions in different scenes. In recent years, biologists have discovered a group of neurons with extreme collision selectivity in the *Drosophila* visual nerve, the lobular plate/lobular column II (LPLC2) neural cluster: it is extreme in that it only responds to objects moving radially from the center of the receptive field of view. Based on this, how to realize responses to objects moving radially from other visual field quadrants is a gap in current algorithmic research. Therefore, to address the problem of detecting object collisions from each visual field quadrant, this paper focuses on studying and modeling the motion localization information generated by T4/T5 cells in the *Drosophila* brain. The LPLC2 collision perception model guided by the attentional mechanism proposed in this paper processes the received motion localization information, relocates the center of the field of view, and calculates the motion of the object along the four basic directions of horizontal and vertical with the new center of the field of view, and then multiplies the outputs of the four quadrants to which the results of the calculations are mapped to achieve the LPLC2 model for collision perception in other quadrants.

In order to fully validate the proposed model, a series of videos-including looming motion scenes from each quadrant with looming motion scenes of objects with different contrasts in multiple quadrants in both simple and complex backgrounds-are experimentally constructed in this paper and compared with the classical LGMD neural network model. The results of the above experiments show that the attention-guided LPLC2 model is able to effectively detect the looming

motion objects from all quadrants, especially when there are multiple looming objects with different contrasts in different quadrants at the same time, the model is more inclined to recognize the looming objects with high contrast. The introduction of the attention guidance mechanism can greatly improve the rationality and robustness of the LPLC2 model.

**KEY WORDS**    *Drosophila* visual system; LPLC2; neural network model; extreme collision selectivity; attentional mechanisms

# 目录

1. 前言 .....	1
1.1 研究背景与意义 .....	1
1.2 论文的主要内容 .....	2
2. 研究现状 .....	3
2.1 基于蝗虫视觉系统的碰撞感知模型 .....	3
2.1.1 LGMD1 神经元 .....	3
2.1.2 LGMD1 模型 .....	3
2.1.3 LGMD2 神经元 .....	4
2.1.4 LGMD2 模型 .....	4
2.2 基于果蝇视觉系统的碰撞感知模型 .....	5
2.2.1 LPLC2 神经元 .....	5
2.2.2 LPLC2 模型 .....	7
3. 计算模型 .....	7
3.1 视网膜层 .....	7
3.2 薄膜层 .....	8
3.3 髓质层 .....	8
3.4 T4/T5 神经元 .....	9
3.5 LPLC2 神经元 .....	10
4. 数值实验 .....	11
4.1 实验视频数据 .....	11
4.2 模型参数设置 .....	13
4.3 实验结果与分析 .....	14
4.3.1 简单背景下注意力引导的 LPLC2 与 LGMD1 模型的对比实验 ...	14
4.3.2 简单背景下多对比度物体迫近实验 .....	18
4.3.3 复杂背景下物体迫近实验 .....	21
5. 总结与展望 .....	24
参考文献 .....	25
致谢 .....	28
附录 .....	29

# 1. 前言

## 1.1 研究背景与意义

在复杂的自然环境中，小到飞虫，大到我们人类，在觅食和归巢的过程中，能够及时并且准确地识别到即将发生的碰撞，是动物识别危险而做出躲避动作的关键。例如：蝗虫可以密集的虫群中飞行数百公里而不发生碰撞；蜜蜂在穿越隧道时表现出的居中行为；螳螂可以在混乱的场景中注意到移动中的小猎物。从不同昆虫种类的碰撞检测神经系统的共同性中学习，可以得到一些有用的启发。当处于碰撞过程中的物体从整个视野的不同区域接近时，接近或迫近的运动会在视网膜上形成边缘扩展模式，逐渐占据甚至覆盖大部分视野。因此，提取边缘的扩张是解决碰撞感知问题的有效方法之一。

经过数千个世纪的进化发展，赋予了动物从脊椎动物如鸽子到无脊椎动物螃蟹，特别是果蝇和蝗虫，它们的复杂视觉系统能够精确提取不同类型的运动模式，让漫游或飞行在复杂的自然环境变得简单。与哺乳动物相比，昆虫可以用相对较少的神经元处理复杂的真实环境。然而，将潜在的机制和回路映射到更高水平行为的神经处理仍然是科学家面临的巨大挑战。利用昆虫视觉神经的灵感，数学神经网络在整个视野碰撞感知方面显示出了它们的优势，即与接近的物体的初始位置无关。其中一个很好的例子是蝗虫，它们通常会成群结队地迁徙。在他们的视觉系统中，小叶巨大运动探测器（LGMDs），即 LGMD1 和 LGMD2，已被确定为具有独特特征的潜在敏感神经元，因此进一步研究建立了数值模型。

果蝇的视觉神经网络也是研究碰撞感知策略的一个重要模型。近年来，随着果蝇视觉神经解剖学的研究飞速进展，学者们受到果蝇视觉视觉的启发，用大量的计算建模工作来模拟果蝇的视觉感知机制，这一类的神经网络模型都有强烈的碰撞选择性——即只对物体的接近做出强烈反应，而对其他运动（如远离或平移）保持沉默。与蝗虫的视觉系统相比，果蝇的视觉系统中的 LPLC2 神经元更是只对来自视野中心的迫近运动进行检测响应。

这种由生物学启发的神经网络模型在一些算力有限的微型机器上具有特别的优势，该类模型的能耗和鲁棒性尤为突出。此外，随着近些年来一些方法的提出，加强了昆虫碰撞检测器在比较复杂的环境的鲁棒性。这些以昆虫为灵感的方法在一定程度上克服了上述问题，为指导制造廉价、快速和可靠的碰撞探

测器提供了建议或解决方案。

## 1.2 论文的主要内容

在果蝇的视觉通路中,最近的生物学研究成果发现了一个神经簇,即 LPLC2 神经簇。经过测试,该神经簇能对直线碰撞过程中由视野中心点发出的黑色的迫近物体做出最强烈的反应,而对来自视野其他象限区域的迫近物体则保持抑制<sup>[2]</sup>。这种极端选择性的视觉神经元由四种对正方向敏感的前突触神经元组成,这些神经元被特别命名为小叶板切向细胞(LPTCs),它们在通过径向运动模式感知来解决碰撞感知问题方面展现出了惊人的潜力和巨大的可能性。

因此,基于确凿而明确的生物学证据,有人从果蝇视觉神经通路中的 LPLC2 得到灵感,提出了一种通过非线性计算运动相关性来塑造其超选择性特征的方法<sup>[23]</sup>。通过模仿 LPLC2 神经元的结构及其突触前小叶板切向细胞(LPTCs)的特殊位置,该模型可以计算每个基本运动方向的对立性,并通过独立的 ON/OFF 通道分别传递亮度的增加和减少。随后输出信息被映射到四个单独的象限区域内,并最终由 LPLC2 神经元进行总结。

但该方法只实现了对视野中心点发出的迫近物体产生反应,而对来自视野其他象限区域的迫近物体则保持抑制,导致了该方法对使用场景有较大的局限性。因此,是否可以通过注意力机制来改变该方法检测的视野中心,重新建立模型以达到对全视野的迫近物体都产生反应,是本文研究的创新研究重点。

本文采用计算机软件生成的模拟视频,在简单背景和复杂背景下对视频的各个象限内的黑圈或白圈扩张的视频信号对模型进行综合性的实验对比。同时与蝗虫启发的 LGMD1 模型,进行实验测试。通过对比实验的结果,证明非线性计算运动相关性来塑造 LPLC2 对碰撞反应具有超选择性,在区分物体的不同运动方式等问题方面是有效的。进一步,本文首次在 LPLC2 模型的基础上引入了注意力机制,实现对在视野的不同象限迫近物体的碰撞感知。

第二章对当前的生物启发的神经网络模型进行一个简单的总结;第三章介绍由注意力引导的 LPLC2 模型的公式以及一些参数取值;第四章介绍实验细节,然后对实验的结果进行分析与总结。第五章是本文总结与研究展望。

## 2. 研究现状

本章节将介绍一些由生物启发的经典的神经网络碰撞感知模型的由来,以及各个模型之间的区别。

### 2.1 基于蝗虫视觉系统的碰撞感知模型

早在 20 世纪 70 年代,生物学家就在生物解剖学上发现了位于蝗虫的视觉大脑小叶神经丛层中的一组大型中间神经元。这些神经元被称为小叶巨运动探测器 (LGMDs)。

#### 2.1.1 LGMD1 神经元

LGMD1 在最初被认为是运动探测器,后来逐渐被认为是迫近物体探测器<sup>[16]</sup>。目前绝大多数研究都集中在 LGMD1 上,该神经元已被证明在能够飞行的蝗虫中发挥主导作用。总的来说, LGMD1 整合了来自不同树突区域之间的视觉信号;它们都产生两种不同基本神经过程:兴奋和抑制。传递神经电路内的神经处理的是这两种类型的神经过程之间的竞争。更准确地说, LGMD1 的激活需要竞争的获胜者是兴奋。此外,下行对侧运动检测器 (DCMD) 是突触后目标神经元与 LGMD1 的一对一连接;该神经元将 LGMD1 生成的尖峰传递给后续的运动控制神经系统,从而产生回避行为。

#### 2.1.2 LGMD1 模型

自 20 世纪 90 年代以来,出现了对碰撞敏感神经元的计算模型。Rind 和 Bramwell 进行了开创性工作,对基于 LGMD1 的神经网络进行建模<sup>[24]</sup>。他们深入地研究了碰撞敏感神经网络中的突触前信号处理机制,并提出了一种解释侧抑制如何在减少时间运动依赖性兴奋方面发挥关键作用的方法;更重要的是,这种机制有效地塑造了 LGMD1 的选择性,以对接近的物体做出强烈的反应。此类信号之间的相互作用决定了 LGMD1 对迫近运动而不是后退和平移运动的特定碰撞选择性。



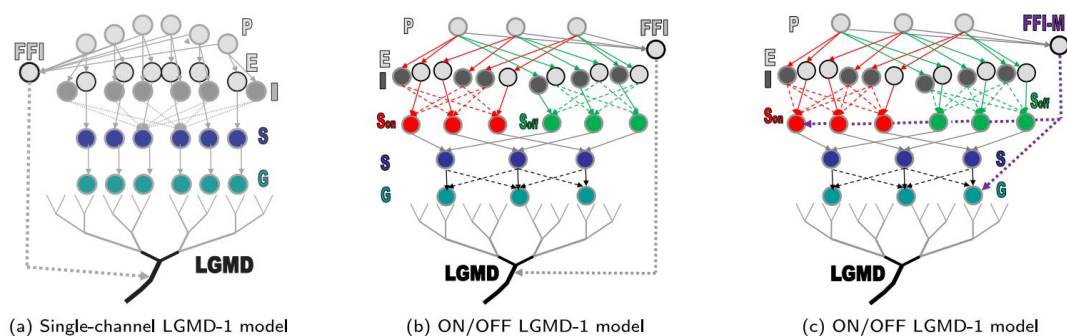


图 2-1 LGMD1 模型<sup>[9]</sup>

### 2.1.3 LGMD2 神经元

LGMD2 也是 LGMD1 的邻居神经元，作为迫近运动的探测器。它与 LGMD1 具有相似的特性，但对运动检测选择性不同。在神经形态学方面，LGMD2 在其突触前区域也有大型扇形树突树。但是，与 LGMD1 相比，LGMD2 中没有发现传递前馈抑制的横向场，而且到目前为止尚未发现 LGMD2 的突触后目标神经元。更重要的是，一项生理学研究证明了蝗虫从青春期到成年期间两种神经元的发育：LGMD2 在缺乏翅膀、主要生活在地面的幼年蝗虫中成熟得更早。所以 LGMD2 有一种特殊的神经特征：它只对明亮背景中迫近的黑暗物体敏感，而对黑暗背景中接近的白色或浅色物体没有反应，这代表了对黑暗背景的偏好<sup>[26]</sup>。

### 2.1.4 LGMD2 模型

对于 LGMD2 的计算建模，到目前为止仅提出了少数研究。直到 2015 年出现了一项开创性的工作：傅沁冰等人提出了一种基于 LGMD2 的视觉神经网络，并且在基于视觉的微型机器人中实现 LGMD2<sup>[27]</sup>。通过 ON 和 OFF 机制的建模，使得模型具有与明暗亮度变化类似的选择性<sup>[28]</sup>。该模型将亮度变化分离到 ON/OFF 并行通道中，并通过与 LGMD1 模型类似的时空计算对兴奋和抑制进行编码，但两个通道具有不同的延迟信息。更具体地说，为了实现 LGMD2 仅对暗物体的特定选择性，ON 通道信息经过更加严格筛分。ON 和 OFF 机制还在每个独立通道中带来不同的时间延迟信息。这种基于 LGMD2 的视觉神经网络的有效性和灵活性已通过自主微型机器人的测试得到验证。生物学家揭示的 LGMD2 神经元的特定功能首次得到建模实现。与所有上述 LGMD1 模型相比，它仅对暗物体的靠近做

出响应，并对亮物体的后退做出短暂响应，表达了该模型对由明到暗亮度变化的偏好。

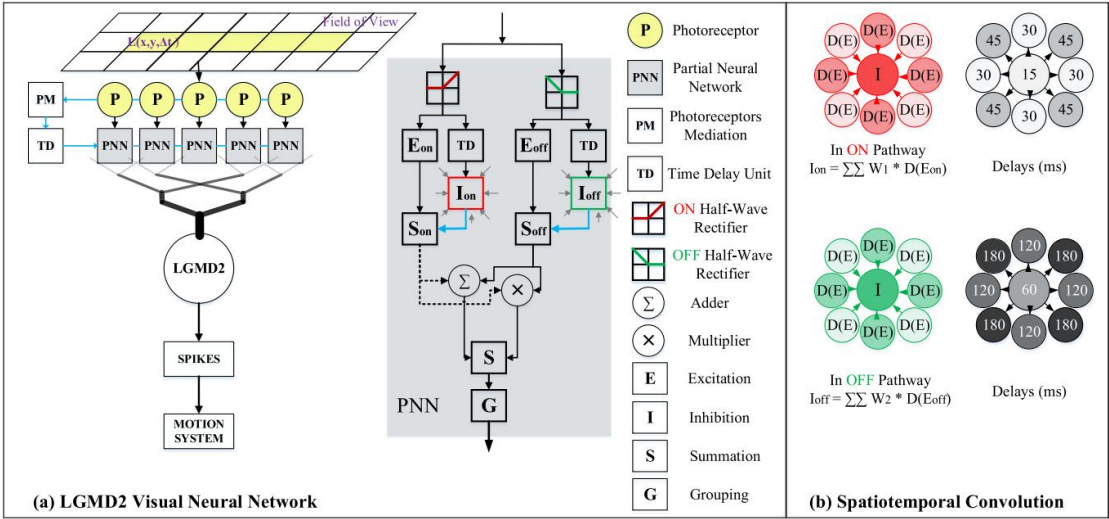


图 2-2 LGMD2 模型<sup>[9]</sup>

## 2.2 基于果蝇视觉系统的碰撞感知模型

在 2017 年，Nathan 等人在果蝇中发现的超选择性迫近检测神经元，称之为小叶板/小叶柱状 II 型 (LPLC2)，LPLC2 神经元是一个由大约 80 个视觉投射神经元组成的群体<sup>[2]</sup>。

### 2.2.1 LPLC2 神经元

Nathan 等人在果蝇中发现的超选择性迫近检测神经元，称之为小叶板/小叶柱状 II 型 (LPLC2)，展示了其选择性是如何通过径向运动对抗来建立的。在果蝇视觉系统中，称为 T4 和 T5 的方向选择性小视野神经元在小叶板中形成空间图，它们各自终止于四个视网膜定位层的其中一个，这样每一层都会对不同基本方向的运动做出反应。单细胞解剖分析表明，LPLC2 十字形初级树突的每个臂在其中一层中分叉，并沿着该层的首选运动方向延伸。

果蝇体内钙成像表明，正如其形状所预测的那样，单个 LPLC2 神经元对从神经元感受野中心发出的向外运动做出强烈反应。每个树突臂还接收局部抑制输入，这些输入定向选择性地用于与激励相反的向内运动。这种径向运动的对抗性产生了兴奋和抑制的平衡，使 LPLC2 对相关运动模式（例如收缩、宽场旋转或平移）没有反应。

如图 2-2，点状线表示视网膜层光感受器接受视觉信号的途径，这些信号首先在薄膜层处理，然后在髓质层中分离成平行的对光流亮度变化具有不同敏感程度的 ON/OFF 通道。然后将 ON 通道中的信号传递给小叶板层通道内的各类(T4s)进行定向运动计算, 而 OFF 通道中的信号传递给小叶层通道内的各类(T5s)进行定向运动计算。其中 ON 通道用实线表示，OFF 通道用虚线表示。蓝色、红色、绿色和紫色区域表示不同类型方向选择神经元。更具体地说，蓝色的神经元对向上运动感兴趣，而紫色的神经元更喜欢向下运动;红线和绿线分别表示对向左和向右运动具有偏好。这些由 T4/T5 中间神经元估计的视觉信号进行结合，并进一步过滤到小叶板切向细胞 LPTCs 进行计算整合<sup>[29]</sup>。在解剖学上，数十个方向选择性的 LPTC 神经元终止于 LPLC2 神经元，最终的 LPLC2 的膜电位决定了其突触后巨纤维是否被激活。

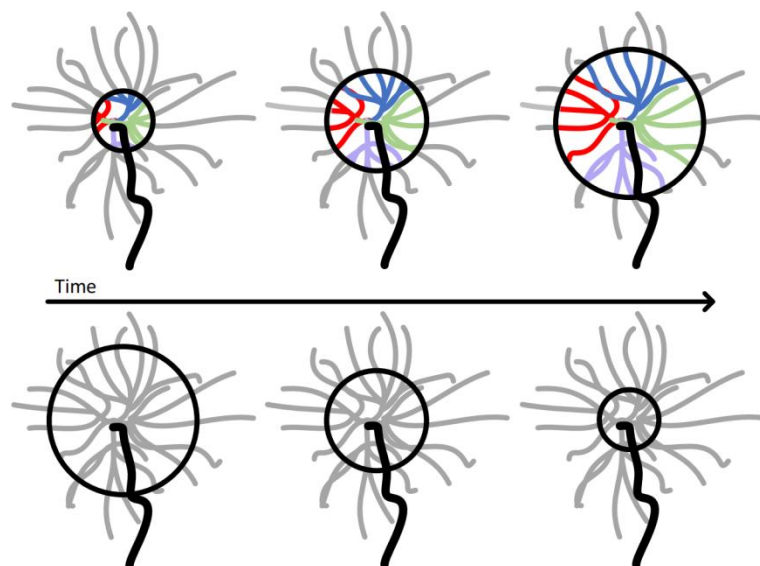


图 2-1 LPTCs 对边缘扩张（上图）和边缘衰退（下图）反应的示意图<sup>[23]</sup>

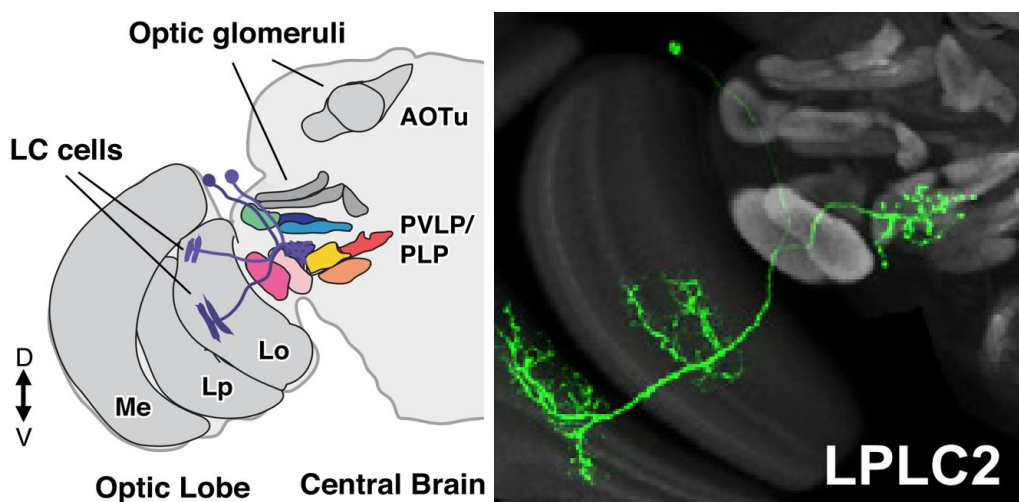


图 2-2 果蝇视觉神经系统示意图<sup>[20]</sup>

### 2.2.2 LPLC2 模型

如图 2-3，首先，光感受器接受两个连续帧的视觉信号，然后获得两帧之间的时间差，然后通过类似高斯的核进行卷积，卷积完进行类似高斯差分算法。然后，卷积的边缘变化被分离成 ON/OFF 平行通道在 ON 通道之后，极化的 T4/T5 中间神经元提取运动估计信息。T4/T5 神经元的运动估计信息会一起作为 LPTC 的输入进行求和。特殊的是，LPTCs 的独特位置被认为会对运动对立相关性产生影响，所以我们提出利用 T5 的运动估计信息，在 ROI 层重新计算 LPTCs 的位置。最后 LPLC2 神经元会以非线性方式计算最终的运动信息。

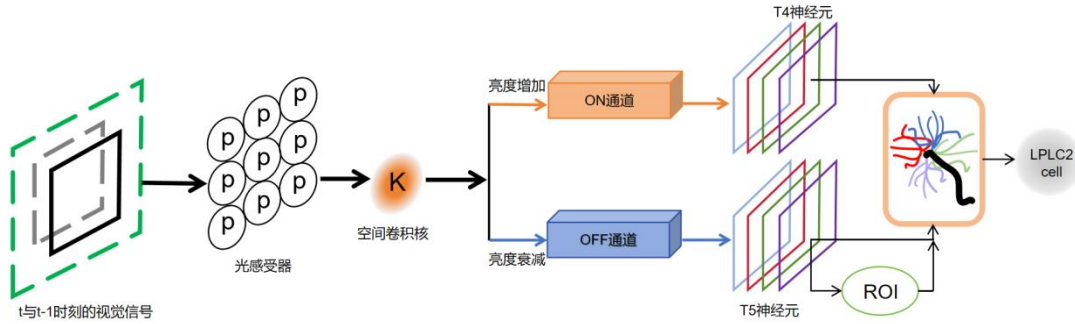


图 2-3 LPLC2 神经网络示意图

## 3. 计算模型

本章节介绍果蝇的 LPLC2 神经元模型的详细数学公式与参数设置，通过数学公式进行系统论述，能够不同象限中逼近运动物体做出响应。该模型是一个模拟生物视觉系统的神经网络，由放置在五个计算神经层的不同类型的神经组成。

### 3.1 视网膜层

视网膜层来自于果蝇的复眼结构，以二维矩阵形式排列的光感受器，其数量对应于输入视觉的分辨率。每个光感受器捕获灰度亮度，然后将其传递给简化的高通滤波器，以获得连续帧之间的亮度变化，可定义为：

$$P(x, y, t) = \sigma_{hp} \cdot (P(x, y, t-1) + L(x, y, t) - L(x, y, t-1)) \quad (1)$$

$$\sigma_{hp} = \tau_i / (\tau_1 + \tau_i) \quad (2)$$

其中  $L(x, y, t)$  为代表每个局部像素对应的灰度亮度数值， $x$  为横坐标， $y$  为纵坐标， $t$  表示时间帧。 $\tau_1$  为一个时间常数， $\tau_i$  为输入的两帧之间的间隔时间。

### 3.2 薄膜层

在薄膜层，为了增强运动边缘的选择性和最大限度地从视觉混乱的环境中传递有用的信息，运用了一种生物启发的机制，被称之为“vDoG”，模拟 LMC 神经元的功能。与传统的 DoG 机制相比，它还展示了 ON 和 OFF 对比度选择性，以适应 ON 和 OFF 通道中的后续处理。vDoG 描述了一种中心-周围对抗结构，中心正高斯和周围负高斯分别代表空间中的兴奋场和抑制场，可定义为：

$$P_e(x, y, t) = \sum_{u=-1}^1 \sum_{v=-1}^1 P(x-u, y-v, t) \cdot G_{\sigma_e}(u, v) \quad (3)$$

$$P_i(x, y, t) = \sum_{u=-2}^2 \sum_{v=-2}^2 P(x-u, y-v, t) \cdot G_{\sigma_i}(u, v) \quad (4)$$

$$G_{\sigma_e}(u, v) = \frac{1}{2\pi\sigma_e^2} \exp\left(-\frac{u^2 + v^2}{2\sigma_e^2}\right) \quad (5)$$

$$G_{\sigma_i}(u, v) = \frac{1}{2\pi\sigma_i^2} \exp\left(-\frac{u^2 + v^2}{2\sigma_i^2}\right) \quad (6)$$

其中  $P_e$  和  $P_i$  分别代表兴奋性信号和抑制性信号， $G$  则为类高斯卷积的内核。

然后具有极性选择性的较窄的兴奋性高斯函数减去较宽的抑制性高斯函数，可定义为：

$$LA(x, y, t) = \begin{cases} |P_e(x, y, t) - P_i(x, y, t)|, & \text{if } P_e(x, y, t) > 0, P_i(x, y, t) > 0 \\ -|P_e(x, y, t) - P_i(x, y, t)|, & \text{if } P_e(x, y, t) < 0, P_i(x, y, t) < 0 \end{cases} \quad (7)$$

### 3.3 髓质层

在髓质层，将来自薄膜层的运动信号分成平行的 ON 和 OFF 通道，分别编码 ON（亮度增加）和 OFF（亮度减小）的响应。存在 ON 和 OFF 半波整流机制，分别通过滤除 ON 和 OFF 路径的负输入和正输入，将运动信息分解为两条平行路径。除此之外，OFF 通路的负输入是信号负的。此外，ON/OFF 路径在当前  $t$  时刻的信号还会受到  $t-1$  时刻信号的影响，计算方法表示如下：

$$\begin{aligned} ON(x, y, t) &= [LA(x, y, t)]^+ + ON(x, y, t-1) \cdot \alpha_1 \\ OFF(x, y, t) &= -[LA(x, y, t)]^- + OFF(x, y, t-1) \cdot \alpha_1 \end{aligned} \quad (8)$$

其中  $[x]^+$  和  $[x]^-$  分别代表  $\max(0, x)$  与  $\min(0, x)$ ， $\alpha_1$  是一个残差常数因子。

对 ON/OFF 路径的信息进行压缩处理，通过以下方法处理：

$$\widehat{ON}(x, y, t) = \sum_{u=-5}^5 \sum_{v=-5}^5 ON(x-u, y-v, t) \cdot G_{\sigma_3}(u, v) \quad (9)$$

$$\widehat{OFF}(x, y, t) = \sum_{u=-5}^5 \sum_{v=-5}^5 OFF(x-u, y-v, t) \cdot G_{\sigma_3}(u, v) \quad (10)$$

$$ON(x, y, t) = \tanh(ON(x, y, t)/(ON'(x, y, t) + C_1)) \quad (11)$$

$$OFF(x, y, t) = \tanh(OFF(x, y, t)/(OFF'(x, y, t) + C_1)) \quad (12)$$

ON/OFF 通道内的信号会受到  $t-1$  时刻的信息残留影响，具体表示如下：

$$DI(x, y, t) = \alpha_3 \cdot R(x, y, t) + (1 - \alpha_3) \cdot R(x, y, t-1) \quad (13)$$

$$\alpha_3 = \tau_i / (\tau_i + \tau_3) \quad (14)$$

其中  $R$  代表为 ON/OFF 通道的压缩信息， $DI$  定义为 ON/OFF 通道里的延迟信息。 $\tau_3$  为一个时间常数。

### 3.4 T4/T5 神经元

当髓质层的信号传入 T4/T4 神经元后，T4/T5 神经元利用压缩信息  $R$  和延迟信息  $DI$ ，以非线性方法对运动进行估计，具体表示如下：

$$ME(x, y, t, d) = DI(x, y, t) \cdot R(x, y + \mu, t) - DI(x, y + \mu, t) \cdot R(x, y, t) \quad (15)$$

其中  $ME$  代表的是 ON 通道中向上运动的运动信息，而 ON 和 OFF 通道的其他主要的运动方向的信息可以由此推断计算得出。

接下来 T4 神经元在  $t$  时刻接受来自 ON 通道的运动信息  $ME$  会与  $t-1$  时刻的运动信息  $ME$  一起产生极化方向信息，同样的 T5 神经元在  $t$  时刻接受来自 OFF 通道的运动信息  $ME$  会与  $t-1$  时刻的运动信息  $ME$  一起产生极化方向信息。

$$T4(x, y, t, d) = \alpha_4 \cdot ME(x, y, t, d) + (1 - \alpha_4) \cdot ME(x, y, t-1, d) \quad (16)$$

$$T5(x, y, t, d) = \alpha_5 \cdot ME(x, y, t, d) + (1 - \alpha_5) \cdot ME(x, y, t-1, d)$$

$$\alpha_4 = \tau_i / (\tau_i + \tau_4) \quad (17)$$

$$\alpha_5 = \tau_i / (\tau_i + \tau_5)$$

其中， $\tau_4$  和  $\tau_5$  分别是 T4 和 T5 神经元的时间常数。 $\alpha_4$  和  $\alpha_5$  表示两个与时间相关

的权重参数,略有不同.当运动信息变量  $d$  落在列表  $[1, 2, 3, 4]$  范围内时,它对应于向右、向左、向下和向上的运动。

接下来,由注意力机制利用 T5 神经元的运动估计信息,来确定 LPTCs 神经元的位置的过程是本次毕业设计的重点,将 T5 神经元运动估计信息进行计算,得出结合了运动路径的 HS 和 VS 系统的局部运动检测器(简称  $LM$ ),接着将其最大模量输出进行最大操作,方法如下:

$$HS(x, y, t) = T5(x, y, t, 1) - T5(x, y, t, 2) \quad (18)$$

$$VS(x, y, t) = T5(x, y, t, 3) - T5(x, y, t, 4)$$

$$\widehat{LM}(x, y, t) = HS^2(x, y, t) + VS^2(x, y, t) \quad (19)$$

$$LM(\hat{x}, \hat{y}, t) = \max_{(x, y) \in (width, height)} \|\widehat{LM}(x, y, t)\|_2 \quad (20)$$

其中,  $LM$  输出的在域  $\Omega(width, height)$  (即在整个视觉空间)的具有位置信息  $(\hat{x}, \hat{y})$  的最大局部运动信号。

接着,用  $(x', y')$  的位置来确定是否需要改变 LPTCs 的位置,过程如下:

$$x^{vc} = \begin{cases} \hat{x}, & \text{if } |x^{vc} - x'| \geq D \\ x^{vc}, & \text{else} \end{cases}, \quad y^{vc} = \begin{cases} \hat{y}, & \text{if } |y^{vc} - y'| \geq D \\ y^{vc}, & \text{else} \end{cases} \quad (21)$$

其中  $x^{vc}$  和  $y^{vc}$  为视觉中心的水平位置与垂直位置,  $D$  为感受域的半径。

### 3.5 LPLC2 神经元

之后,在被 LPLC2 神经元汇总之前,根据  $x^{vc}$  和  $y^{vc}$  确定的感受野位置,将具有方向选择性的 T4/T5 神经元的各个方向的运动估计信息传递到他们的突触后神经元,即 LPTC 神经元,过程如下:

$$Motion(t, d) = \sum_{x=x^{vc}-D}^{x^{vc}+D} \sum_{y=y^{vc}-D}^{y^{vc}+D} (w_1 \cdot T4(x, y, t, d)^{\beta_1} + (w_2 \cdot T5(x, y, t, d)^{\beta_2}) \quad (22)$$

最后,提取四个基本方向的运动信息,由 LPLC2 神经元进行总结。因此,在随后收集某一时刻的水平和垂直运动信息。这里,向右和向下运动方向的数值定义为正值,其他方向的数值定义为负值。总结为:

$$\begin{aligned} H(t) &= Motion(t, 1) - Motion(t, 2) \\ V(t) &= Motion(t, 3) - Motion(t, 4) \end{aligned} \quad (23)$$

水平运动和垂直运动然后通过下面的方程映射到由  $x^{vc}$  和  $y^{vc}$  为原点,重新确定的四个象限,过程如下:

$$\begin{aligned}
Q(t,1) &= g(H(t)) + g(-V(t)) \\
Q(t,2) &= g(-H(t)) + g(-V(t)) \\
Q(t,3) &= g(-H(t)) + g(V(t)) \\
Q(t,4) &= g(H(t)) + g(V(t))
\end{aligned} \tag{23}$$

其中， $Q(t,i)$  表示第  $i$  象限内的区域输出， $g(x)$  是一个近似的高斯误差线性单元激活函数，用于激活运动。

然后 LPLC2 以非线性的方式总结区域信息：

$$LPLC2(t) = Q(t,1) \cdot Q(t,2) \cdot Q(t,3) \cdot Q(t,4) \tag{24}$$

## 4. 数值实验

### 4.1 实验视频数据

本文的实验是利用视频中的模拟迫近或平移运动进行运动检测，迫近视频分三种，第一种是计算机合成的简单无噪声背景的模拟迫近运动的视频，第二种是多种不同对比度的物体从不同象限的模拟迫近运动视频，第三种是复杂场景的迫近运动视频。

第一部分是由计算机生成的模拟运动视频是无噪声的简单场景。该部分模拟了物体从中心和四个不同象限的迫近运动，其中 (a)–(e) 是在黑色背景下的白色圆形物体的迫近运动，(f) 是白色方块的平移运动，(g)–(k) 是在白色背景下的黑色圆形物体的迫近运动，(l) 是黑色方块的平移运动。



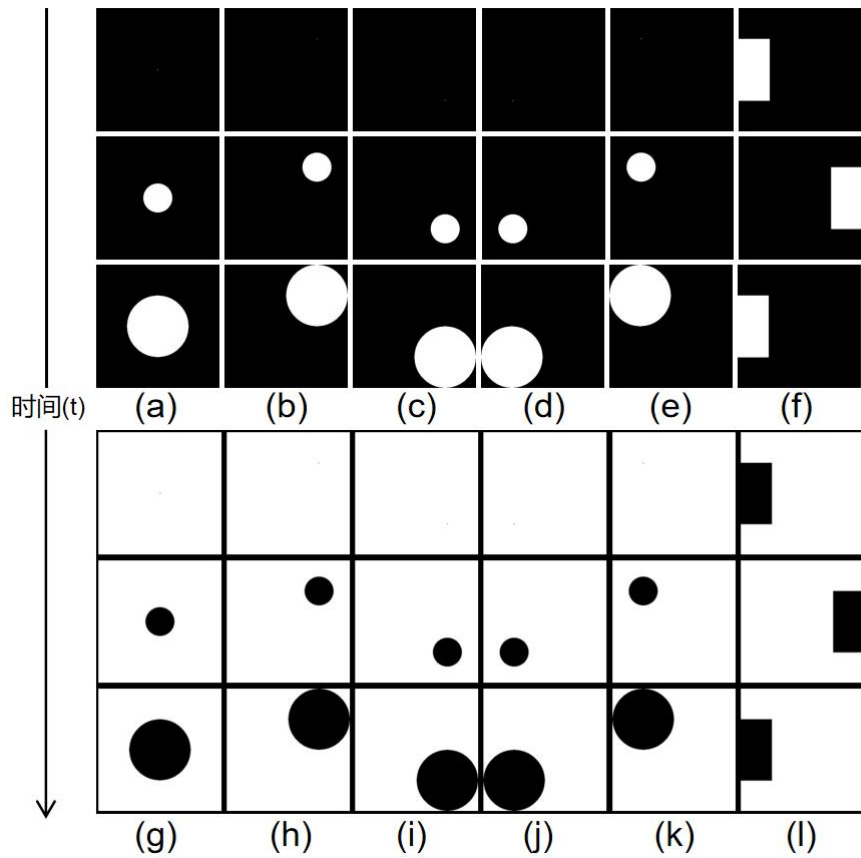


图 4-1 简单背景下的实验视频三个阶段的快照

第二部分是由计算机生成的模拟运动视频是无噪声的简单场景。该部分模拟了多个不同对比度物体从不同象限的迫近运动，其中(a)–(c)是在白色背景下的两个不同对比度(黑色与灰色)的圆形从两个不同的象限的迫近运动，其中(d)–(f)是在白色背景下的三个不同对比度(黑色、灰色和灰白色)的圆形从三个不同的象限的迫近运动。

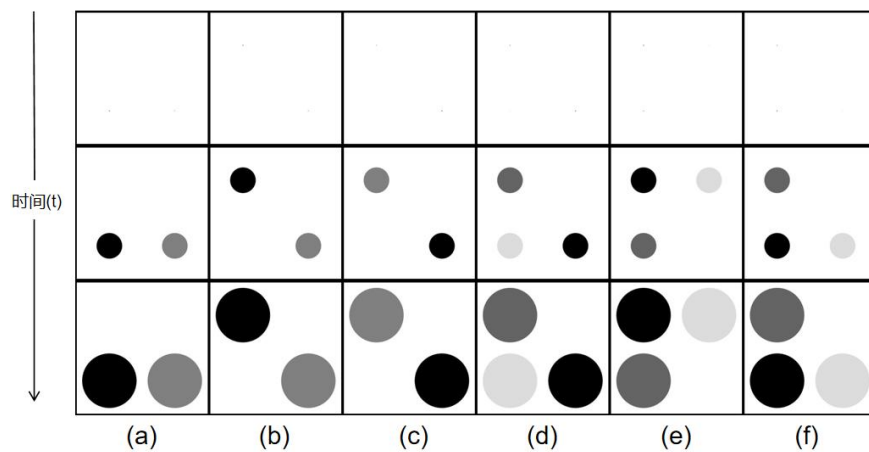


图 4-2 不同灰度值迫近运动实验视频的快照

第三部分是在一个有噪声复杂背景下的测试场景。该部分与第一部分一样模拟了物体从中心和四个不同象限的迫近运动。其中(a)–(c)是在自然背景下的黑色圆形在各个象限迫近运动，(d)和(f)是自然背景下的黑色和灰色圆形在两个不同象限迫近运动。

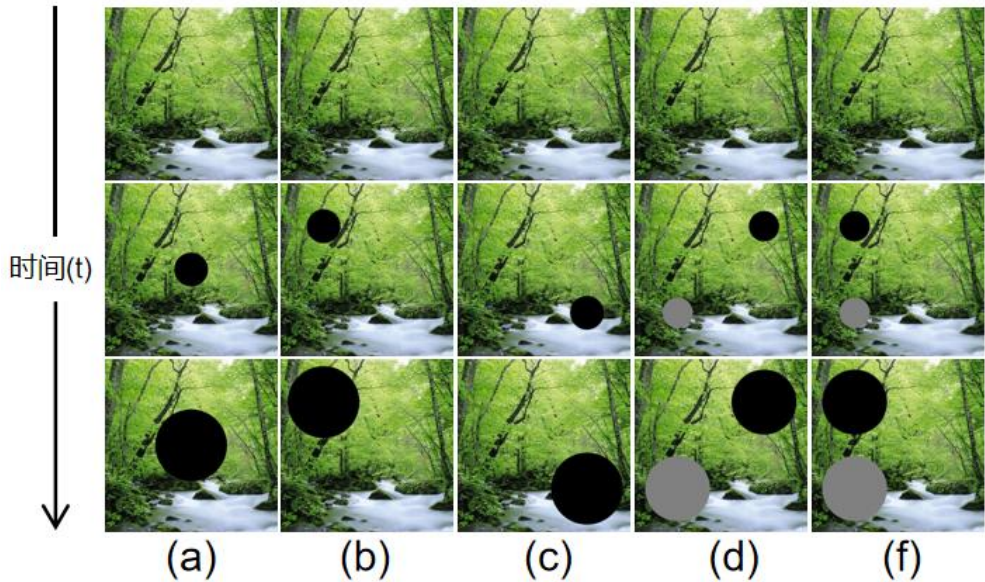


图 4-3 在复杂背景迫近运动实验视频的快照

#### 4.2 模型参数设置

表 4-1 展示了在第三章所介绍的计算模型中的一些参数的数值以及含义。以下参数是经过生物学实验得出，该神经网络模型为前馈结构，无需训练调整参数。

表 4-1 参数设置

参数	描述	数值
$\alpha_1$	ON/OFF 通道残差系数	0.1
$\{\beta_1, \beta_2\}$	T4/T5 神经元反应指数	$\{0.9, 0.5\}$
$\mu$	运动感受器间距	1
$\sigma$	卷积核的标准差	$\{1, 2\}$
$\sigma_2$	径向空间偏差分布的标准差	1
$\sigma_3$	对比度通道的标准差	5
$\tau_i$	两帧之间的时间间隔	1000/60(ms)
$\tau_2$	对比通道时间常数	500(ms)
$\tau_3, \tau_4, \tau_5$	ON/OFF 通道中的时间常数	30(ms)
$w_1, w_2$	ON/OFF 通道中的权重因子	1
$w_c$	对比度通道的加权系数	1
D	感受域半径	50

### 4.3 实验结果与分析

为了验证引入了注意力机制后的 LPLC2 模型是否能检测来自不同象限的物体迫近运动, 以及对不同对比度的物体迫近运动的偏好, 设计了一下几组实验。所有实验视频分辨率均为 400\*400, 采样率为 60hz。

#### 4.3.1 简单背景下注意力引导的 LPLC2 与 LGMD1 模型的对比实验

这一组实验选取了图 4-1 中的视频进行实验, 其背景均为黑色或白色, 无其他噪声, 物体迫近运动的边缘对比度高, 可以较好的展现两个模型之间的区别与引入注意力机制后的改进。

##### (1) 视野中心的迫近运动:

如图 4-4, (a) 为黑色圆形物体在视野中心迫近时 LGMD1 模型, LPLC2 模型和注意力引导的 LPLC2 模型对其运动的反应, (b) 为白色圆形物体在视野中心扩张时 LGMD1 模型, LPLC2 模型和注意力引导的 LPLC2 模型对其运动的反应。可以看出三个模型都对视野中心的迫近运动产生了反应, 但 LPLC2 对黑色圆形物体的迫近运动反应更为强烈。注意力引导的 LPLC2 模型对物体迫近运动到超过感受域时, 不再对物体的迫近运动产生反应。

##### (2) 视野中心的后退运动:

如图 4-5, (a) 为黑色圆形物体在视野中心后退时 LGMD1 模型, LPLC2 模型和注意力引导的 LPLC2 模型对其运动的反应, (b) 为白色圆形物体在视野中心后退时 LGMD1 模型, LPLC2 模型和注意力引导的 LPLC2 模型对其运动的反应。可以看出 LGMD1 模型对后退运动也产生了反应, 而 LPLC2 模型和注意力引导的 LPLC2 模型对后退运动几乎不产生反应。

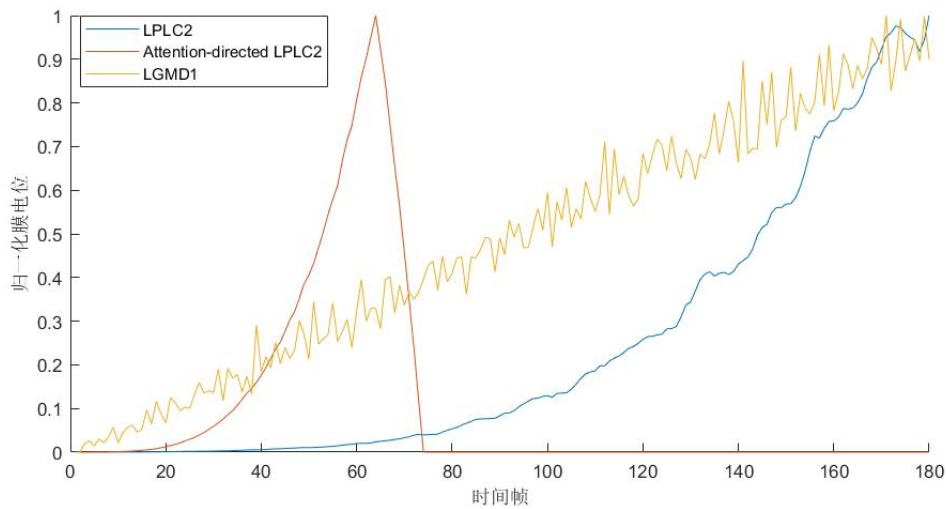
##### (3) 各象限的迫近运动:

如图 4-6 与图 4-7, 分别展示了黑色圆形物体与白色圆形物体在四个象限中心开始迫近运动的各个模型反应, 可以看出 LGMD1 模型与注意力引导的 LPLC2 模型对其运动的反应。同样的, 注意力引导的 LPLC2 模型对物体迫近运动到超过感受域时, 不再对物体的迫近运动产生反应。而 LPLC2 模型对非视野中心的迫近运动不产生反应。

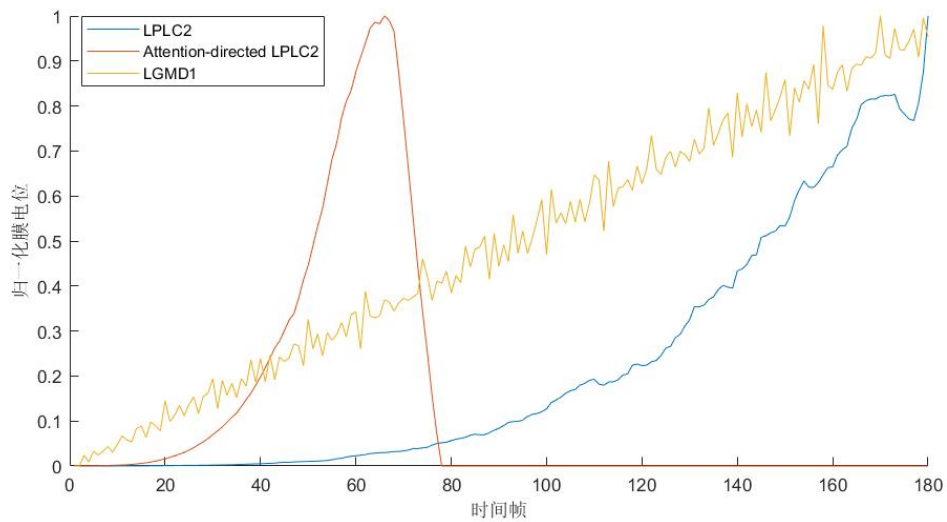
##### (4) 平移运动:

如图 4-8 与图 4-9 分别展示了黑色方形物体与白色方形物体平移运动的实验结果。可以看出，LGMD1 对平移运动会产生强烈的反应，而两种 LPLC2 模型均对平移运动基本不产生反应。

对比(1)(2)(3)(4)的实验结果，可以确定建立的注意力引导的 LPLC2 计算模型在原 LPLC2 计算模型基础上，改进了对各个象限的迫近运动的检测，并取得了较好的效果。

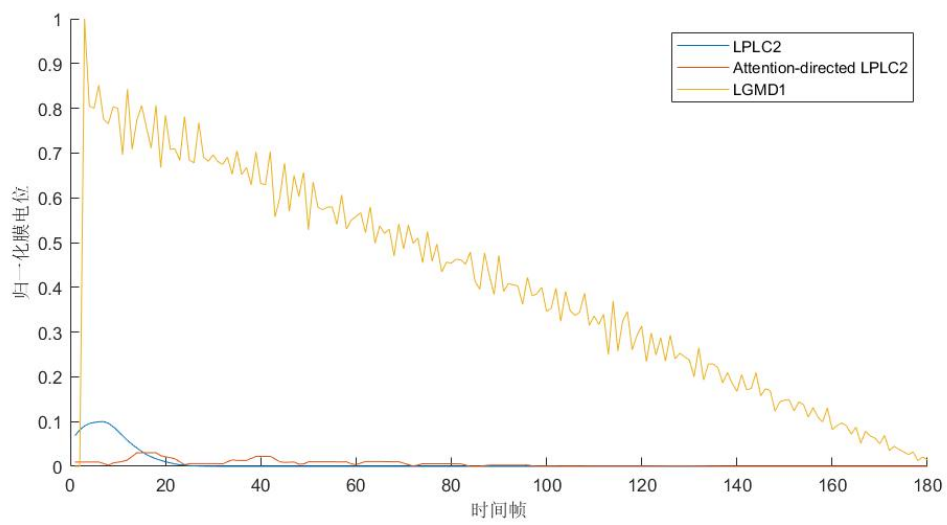


(a) 黑色圆形物体在视野中心迫近运动反应

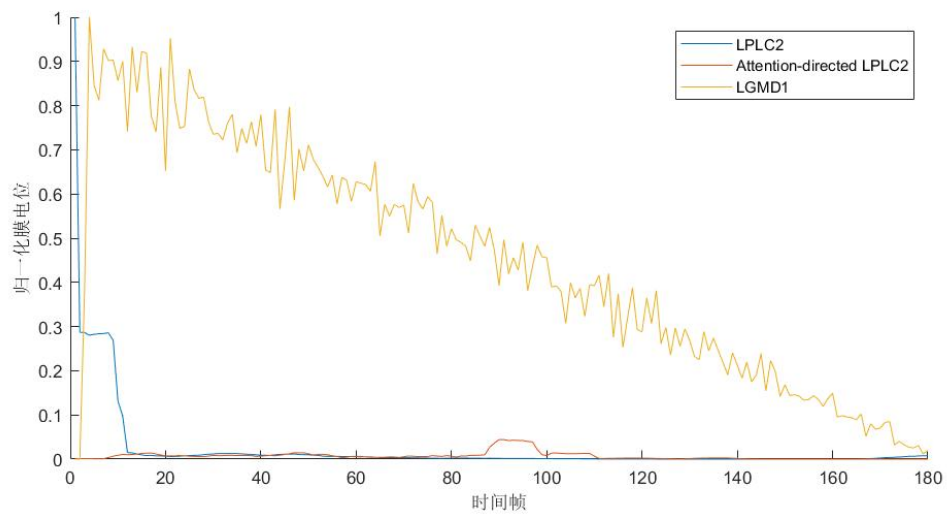


(b) 白色圆形物体在视野中心迫近运动反应

图 4-4 三种模型对不同对比度程度的物体在视野中心迫近运动反应



(a) 黑色圆形物体在视野中心后退运动反应



(b) 白色圆形物体在视野中心后退运动反应

图 4-5 三种模型对不同对比度程度的物体在视野中心后退运动反应

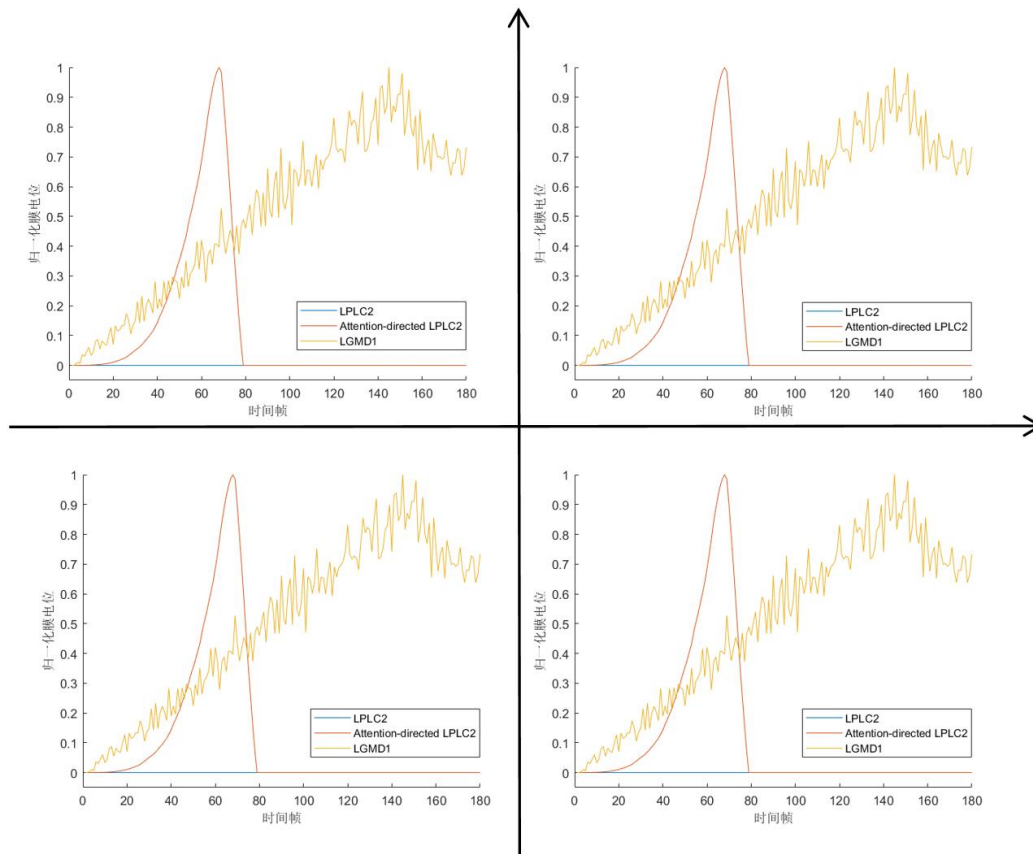


图 4-6 对黑色物体从各个象限逼近的反应

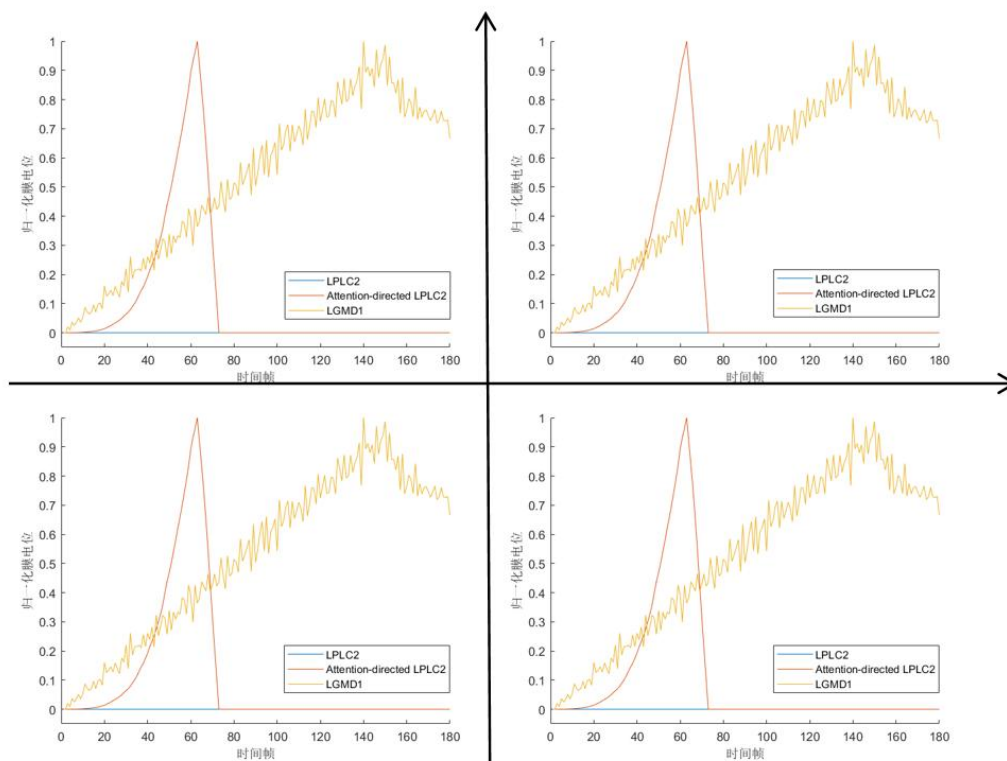


图 4-7 对白色物体从各个象限逼近的反应

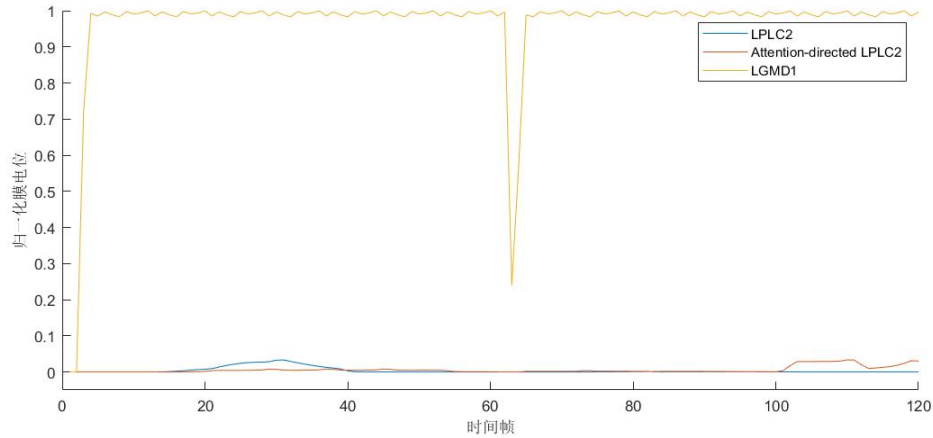


图 4-8 对黑色物体平移运动的反应

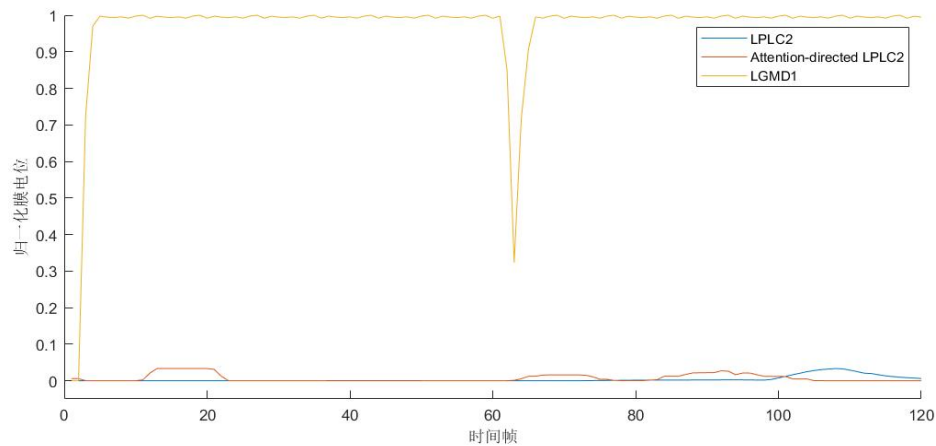


图 4-9 对白色物体平移运动的反应

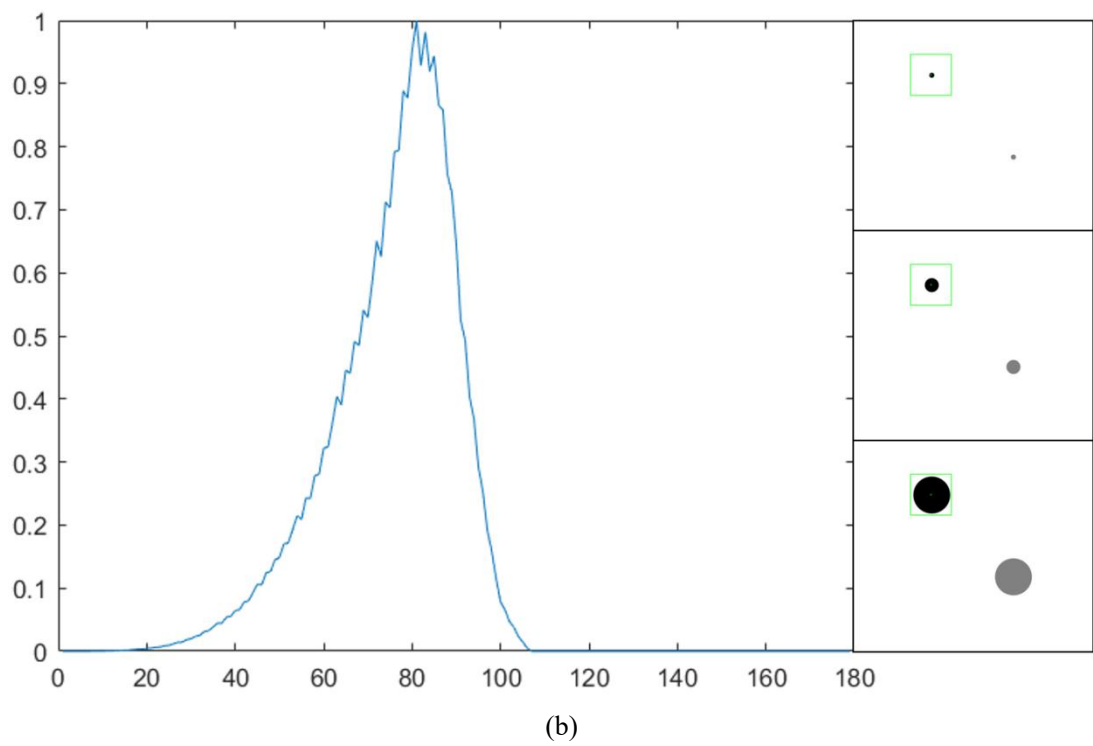
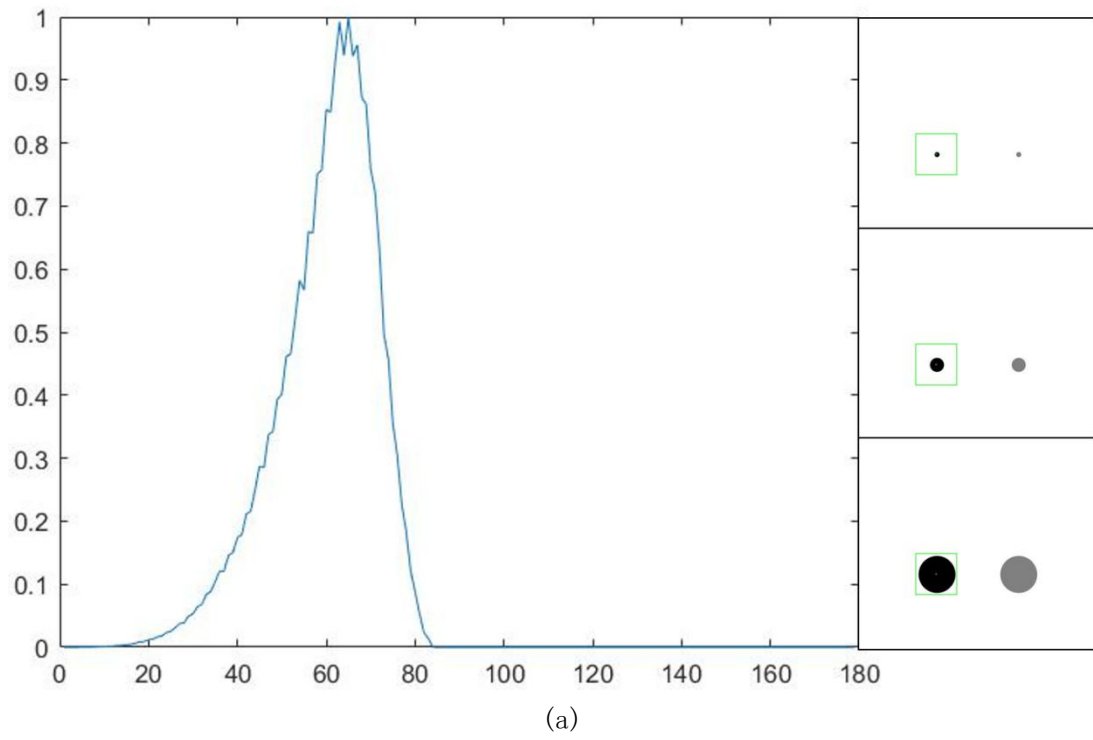
#### 4.3.2 简单背景下多对比度物体迫近实验

本节实验使用了图 4-2 中 6 组在简单环境下不同对比度的圆形物体从各个感受野象限的迫近运动，探讨在受到其他物体干扰的情况下，注意力引导的 LPLC2 计算模型的效果。

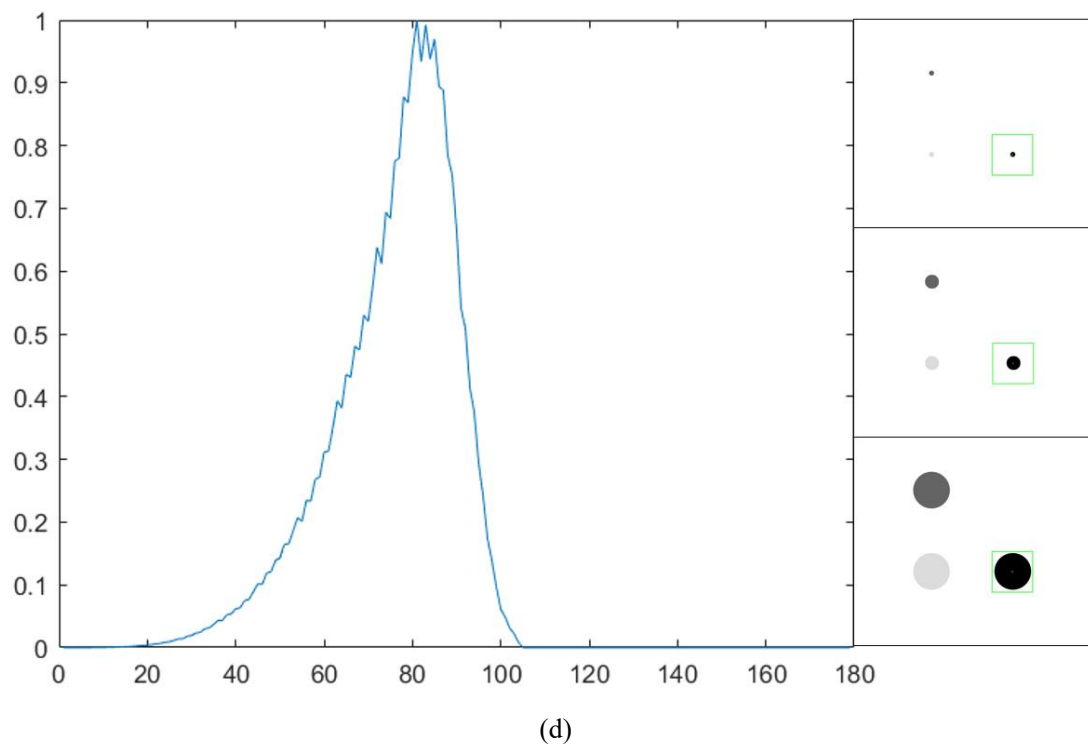
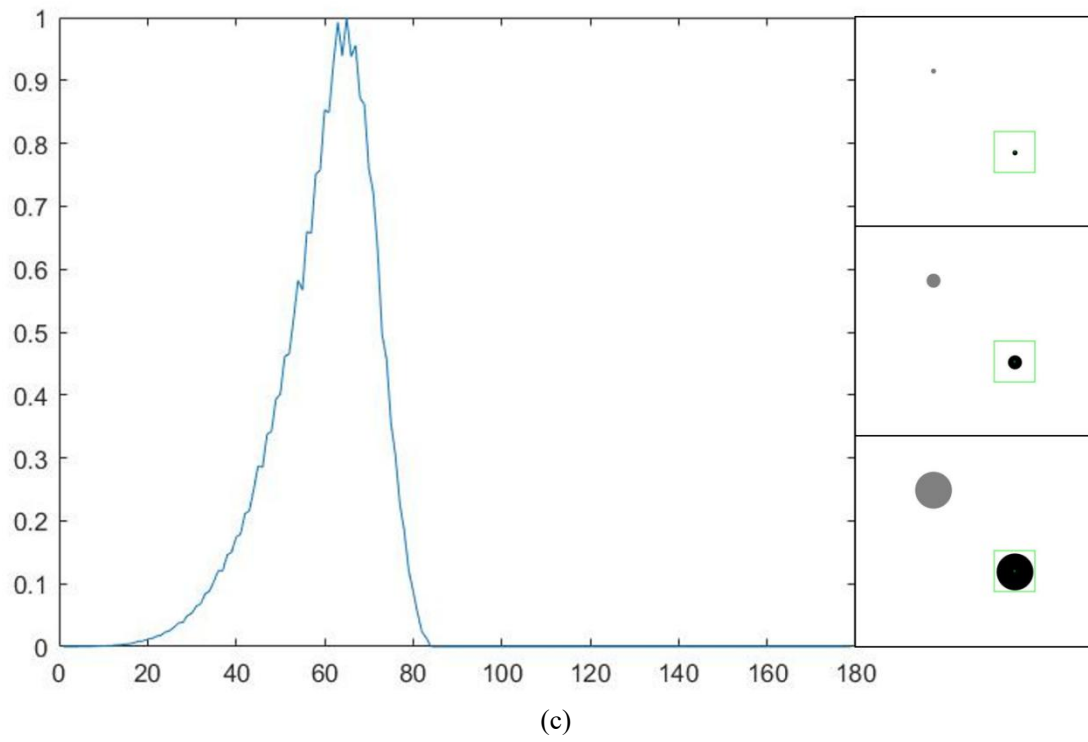
如图 4-10 中，(a)–(f) 六个子图对应了图 4-2 中六个实验视频，实验视频为在简单背景下两个或三个不同对比度的圆形物体从各个象限进行迫近运动。实验结果由模型对实验视频中物体碰撞检测产生的膜电位变化与经过注意力引导的 LPLC2 感受域的变化组成。

分析这六张实验结果，可以发现注意力引导的 LPLC2 计算模型对较暗物体的迫近运动的检测具有选择性。在多种对比度的物体同时靠近时，模型会优先检测较暗物体的迫近运动。当感受域内的物体扩大到超过 LPLC2 感受域时，模型不再检测该物体的碰撞运动。









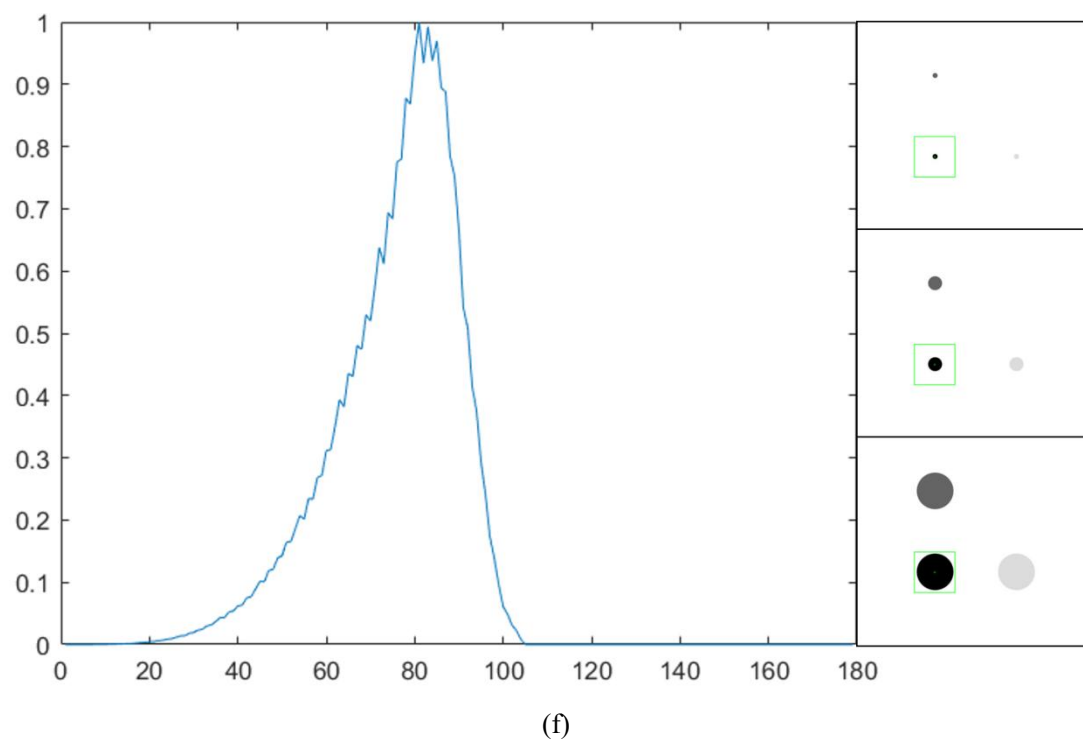
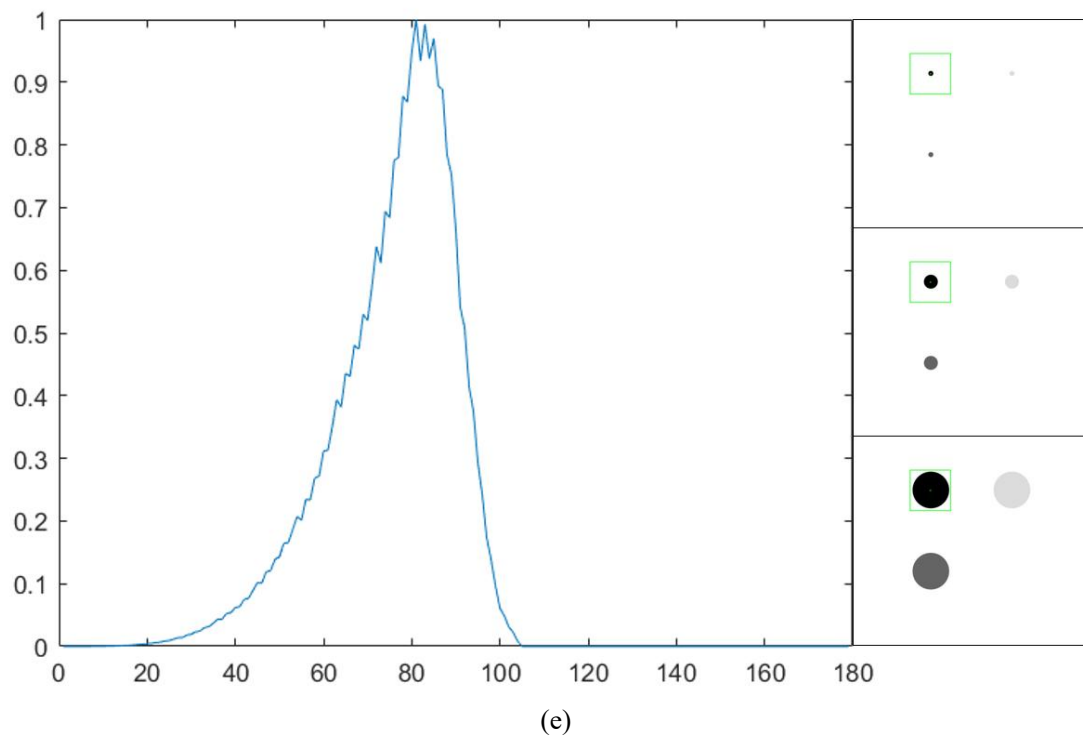


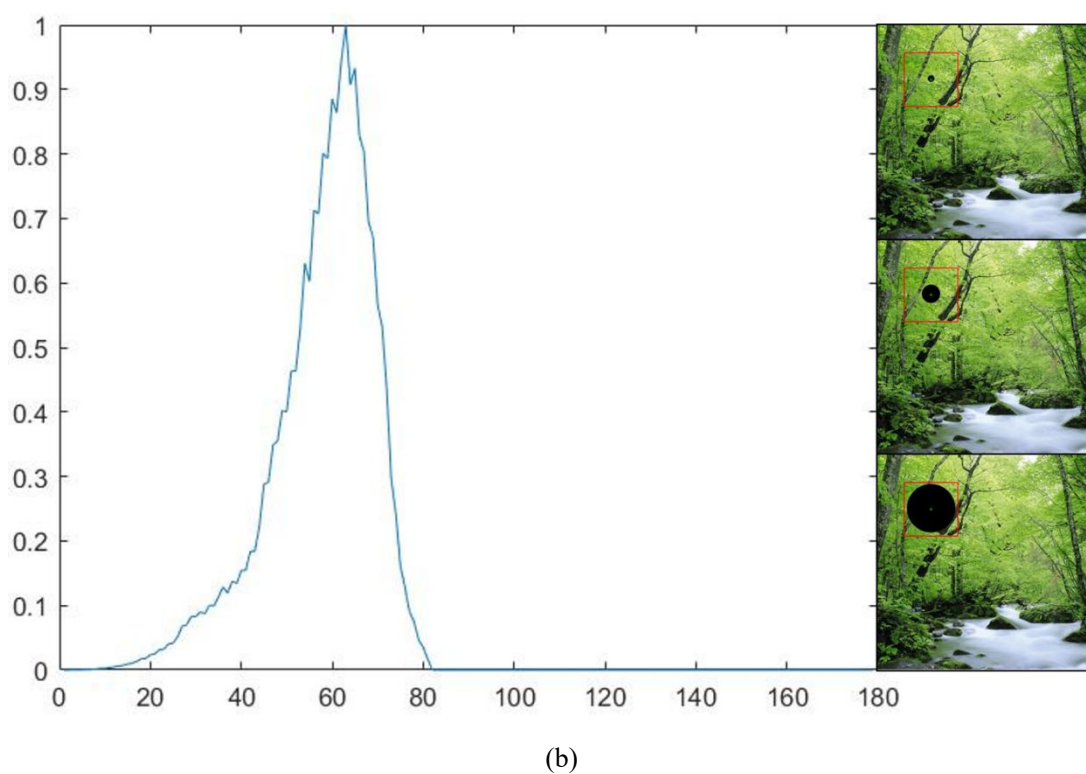
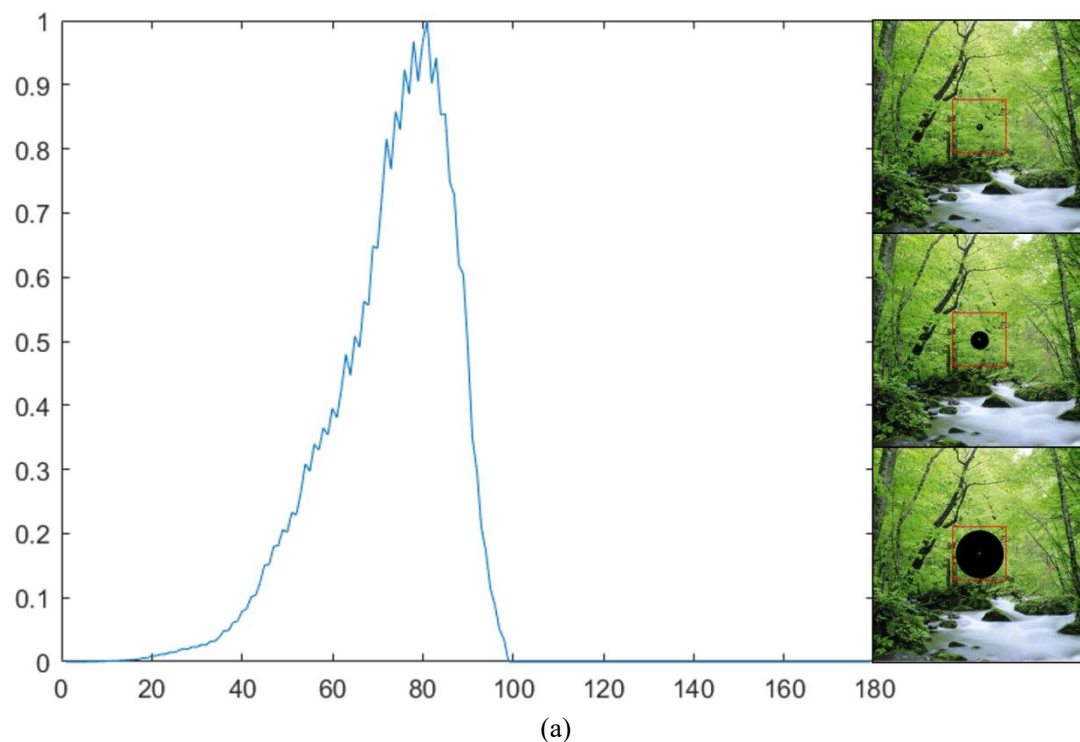
图 4-10 简单背景下多个物体的碰撞检测

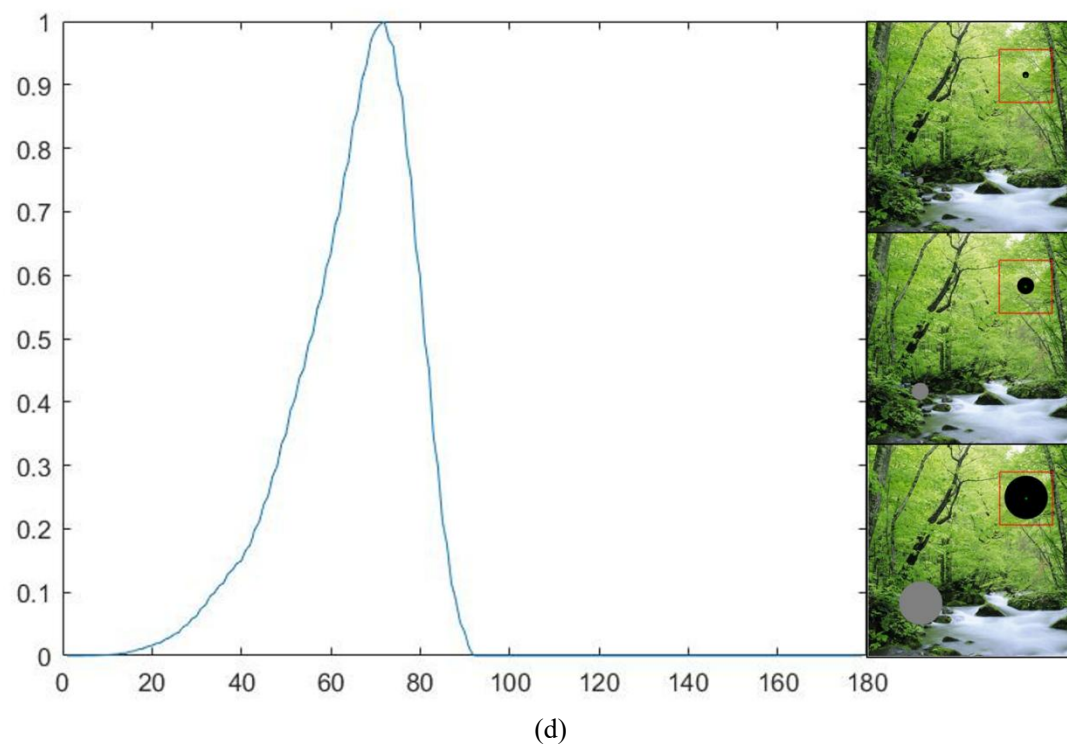
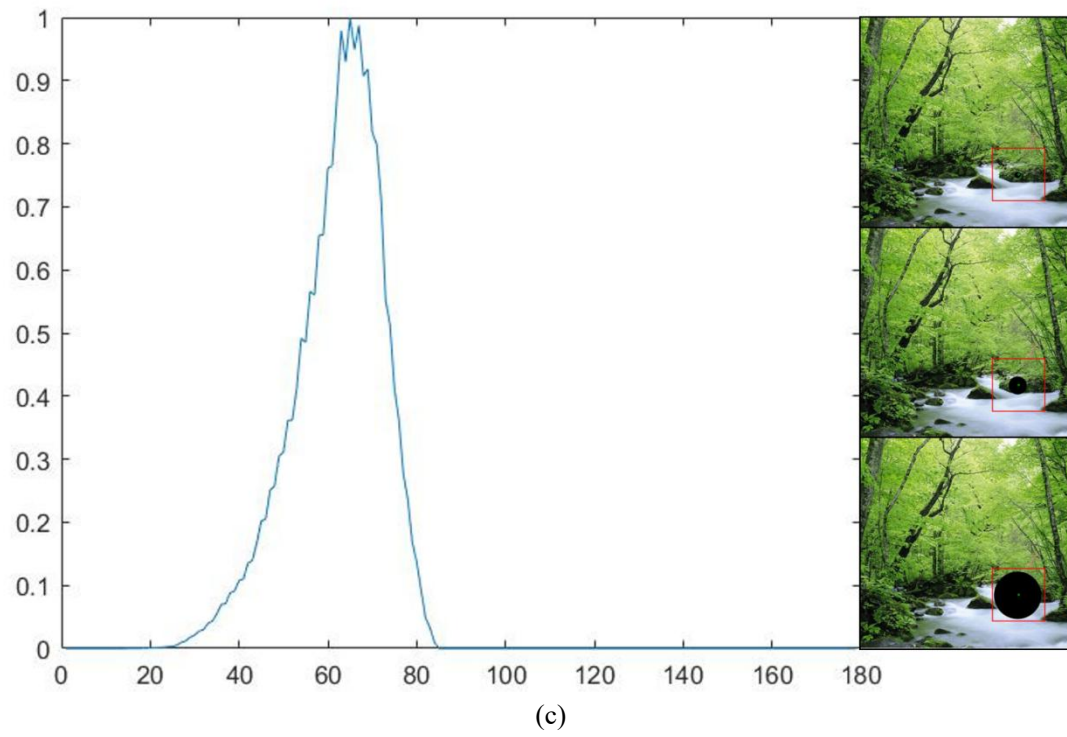
### 4.3.3 复杂背景下物体迫近实验

如图 4-11 中, (a)–(e) 五个子图对应了图 4-3 中五个实验视频, 实验视频为在复杂背景下一个或两个不同对比度的圆形物体从各个象限进行迫近运动。实验

结果由模型对实验视频中物体碰撞检测产生的膜电位变化与经过注意力引导的 LPLC2 感受域的变化组成。

分析这五张实验结果，可以发现注意力引导的 LPLC2 计算模型在复杂背景下依然能检测来自不同象限物体的迫近运动，且对较暗物体的迫近运动的检测仍具有选择性，进一步验证了该模型的鲁棒性。





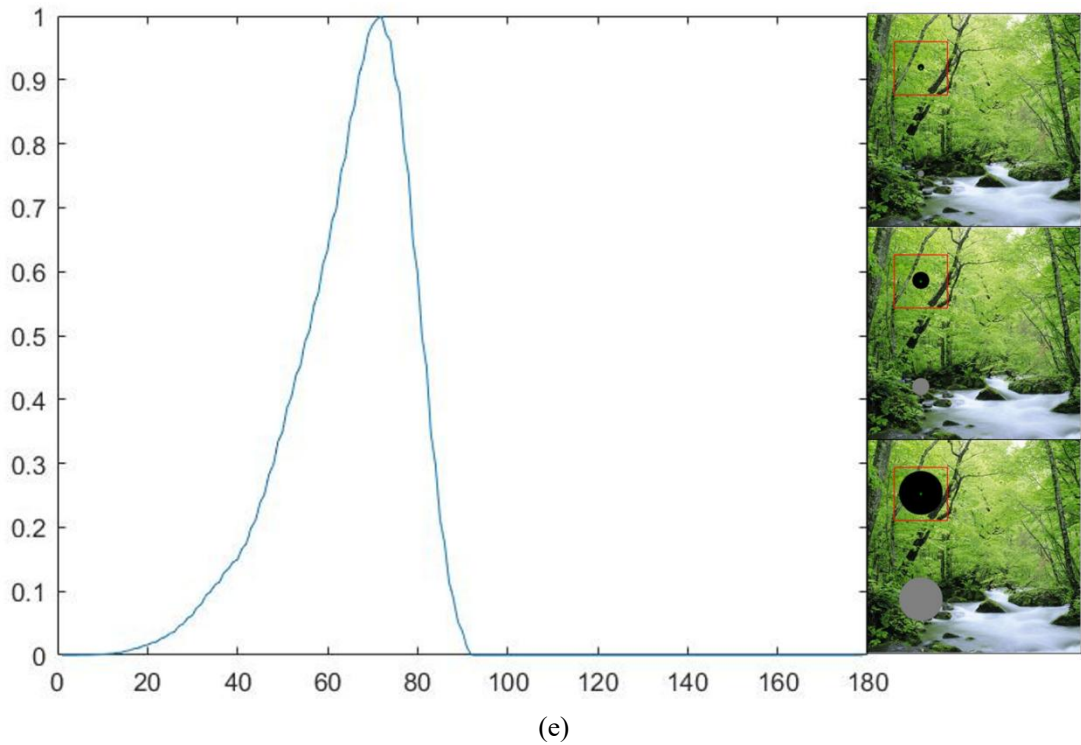


图 4-11 复杂背景下一个或多个物体的碰撞检测

## 5. 总结与展望

综上所述，该模型受到果蝇视觉神经系统的启发，本文构建了一个非线性计算数值模型来塑造由注意力引导的 LPLC2 神经网络模型。与用于碰撞感知的整个射频敏感 LGMD1 神经网络相比，该方法具有超选择性的特点，只对物体的碰撞运动反应，对于物体远离、平移等运动不产生反应。与一般的 LPLC2 神经网络模型相比，该方法能对感受野其他象限的径向运动产生反应。

在本文的模型实验中，我选取了三种场景验证模型，来探究注意力引导的 LPLC2 神经网络模型的碰撞检测能力。在简单场景和复杂场景下的来自不同象限的迫近运动物体，该模型均能检测出。且当多个不同对比度的物体从多个象限进行迫近运动时，该模型对较暗的迫近物体具有选择性检测。

通过实验结果表明，本文构建的由注意力引导的 LPLC2 神经网络模型，可以在简单和复杂环境下很好的检测出来自各个象限迫近运动的物体，注意力机制对迫近运动感知模型在检测不同对比度的迫近运动的物体中有重要作用，且能够提升模型在复杂环境下的鲁棒性。同时，该模型在在复杂环境下对明暗程度的偏好性与时间复杂度上还有优化的空间，以便更好地对迫近物体进行检测反应。

## 参考文献

- [1] Borst, A., Haag, J., & Mauss, A. S. (2020). How fly neurons compute the direction of visual motion[J]. *Journal of Comparative Physiology A*, 206, 109–124.
- [2] Klapoetke, N. C., Nern, A., Peek, M. Y., Rogers, E. M., Breads, P., Rubin, G. M., et al. (2017). Ultra-selective looming detection from radial motion opponency[J]. *Nature*, 551, 237–241.
- [3] Haag, J., Arenz, A., Serbe, E., Gabbiani, F., & Borst, A. (2016). Complementary mechanisms create direction selectivity in the fly[J]. *ELife*, 5, 1–15.
- [4] Joesch, M., Schnell, B., Raghu, S. V., Reiff, D. F., & Borst, A. (2010). ON and OFF pathways in drosophila motion vision[J]. *Nature*, 468, 300–304.
- [5] Jones, P. W., & Gabbiani, F. (2010). Synchronized neural input shapes stimulus selectivity in a collision-detecting neuron[J]. *Current Biology*, 20, 2052–2057.
- [6] Badwan, B. A., Creamer, M. S., Zavatone-Veth, J. A., & Clark, D. A. (2019). Dynamic nonlinearities enable direction opponency in drosophila elementary motion detectors[J]. *Nature Neuroscience*, 22, 1318–1326.
- [7] Maisak, M. S., Haag, J., Ammer, G., Serbe, E., Meier, M., Leonhardt, A., et al. (2013). A directional tuning map of drosophila elementary motion detectors[J]. *Nature*, 500, 212–216.
- [8] Vogt, N., & Desplan, C. (2007). The first steps in drosophila motion detection[J]. *Neuron*, 56, 5–7.
- [9] Q. Fu, Motion perception based on ON/OFF channels: A survey[J]. *Neural Netw.* 165 (2023) 1–18.
- [10] Q. Fu, H. Wang, C. Hu, & S. Yue, (2019). Towards computational models and applications of insect visual systems for motion perception: A review[J]. *Artificial Life*, 25, 263–311.
- [11] Q. Fu, C. Hu, T. Liu, and S. Yue. Collision selective lgmds neuron models research benefits from a vision-based autonomous micro robot[J]. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp.

- 3996–4002.
- [12] Q. Fu and S. Yue. Modelling drosophila motion vision pathways for decoding the direction of translating objects against cluttered moving backgrounds[J]. *Biological Cybernetics*, vol. 114, no. 4, pp. 443–460, 2020.
  - [13] D.Regan and K.Beverley. Looming detectors in the human visual pathway[J]. *Vision Research*, vol. 18, no. 4, pp. 415–421, 1978.
  - [14] J.R. Kohn, S. L. Heath, and R. Behnia. Eyes matched to the prize: The state of matched filters in insect visual circuits[J]. *Frontiers in Neural Circuits*, vol. 12, 2018.
  - [15] B.Zhou, Z.Li, S.Kim, J.Lafferty, and D. A. Clark. Shallow neural networks trained to detect collisions recover features of visual loomselective neurons[J]. *eLife*, vol. 11, p. e72067, 2022.
  - [16] M.O’Shea and J.L.Williams. The anatomy and output connection of a locust visual interneurone; the lobular giant movement detector (lgmd) neurone[J]. *Journal of comparative physiology*, vol. 91, pp. 257–266, 1974.
  - [17] C.Hu, F. Arvin, C. Xiong, and S. Yue. Bio-inspired embedded vision system for autonomous micro-robots: The lgmd case[J]. *IEEE Transactions on Cognitive and Developmental Systems*, vol. 9, no. 3, pp. 241–254, 2017.
  - [18] F. Gabbiani, H.G. Krapp, and G. Laurent. Computation of object approach by a wide-field, motion-sensitive neuron[J]. *Journal of Neuroscience*, vol. 19, no. 3, pp. 1122–1141, 1999.
  - [19] H.Sun and B.J.Frost.Computation of different optical variables of looming objects in pigeon nucleus[J]. *Nature neuroscience*, vol. 1, no. 4, pp. 296–303, 1998
  - [20] Nathan C. Klapoetke,ljoscha Nern. A functionally ordered visual feature map in the Drosophila brain[J]. *Neuron*,vol.110 p.1700-1711.2023
  - [21] M. Wu, A. Nern, W. R. Williamson, M. M. Morimoto, M. B. Reiser, G. M. Card, and G. M. Rubin.Visual projection neurons in the drosophila lobula link feature detection to distinct behavioral programs[J]. *eLife*, vol. 5, p. e21022, 2016.
  - [22] F. Gabbiani, H. G. Krapp, C. Koch, and G. Laurent.Multiplicative computation

- in a visual neuron sensitive to looming[J]. *Nature*, vol. 420, pp. 320–324, 2002.
- [23] Mu Hua et al. Shaping the Ultra-Selectivity of a Looming Detection Neural Network from Non-linear Correlation of Radial Motion[J]. *IJCNN*, 2022
- [24] F. C. Rind and D. I. Bramwell. Neural network based on the input organization of an identified neurone signaling impending collision[J]. *J. Neurophysiol* , vol. 75, no. 3, pp. 967–985, 1996.
- [25] Iida, F. Book review: Flying insects and robots[J]. *Artificial Life*, 18, 125–127, 2016.
- [26] J. Sztarker and F. C. Rind. A look into the cockpit of the developing locust: looming detectors and predator avoidance[J]. *Developmental neurobiology*, vol. 74, no. 11, pp. 1078–1095, 2014.
- [27] Fu, Q., Yue, S., & Hu, C. Bio-inspired collision detector with enhanced selectivity for ground robotic vision system[J]. In E. R. Hancock, Richard C. Wilson, & W. A. P. Smith (Eds.), *British Machine Vision Conference* pp.1–13, 2016.
- [28] Fu, Q., & Yue, S. Modelling LGMD2 visual neuron system[J]. In *Proceedings of the 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing*. pp. 1–6, 2015.
- [29] Q. Fu, C. Hu, J. Peng, F. C. Rind, and S. Yue. A robust collision perception visual neural network with specific selectivity to darker objects[J]. *IEEE Transactions on Cybernetics*, vol. 50, no. 2, pp. 5074–5088, 2020.



## 致谢

耗时近四个月，我的本科论文即将完成，论文的完成也将作为我本科学习生涯画上完美的句号。这一毕业设计的顺利完成，离不开老师、师兄和同学的帮助与支持，我想借此机会向所有在我学业和生活中给予关心和支持的人表示最诚挚的感谢。

感谢广州大学为我提供了优质的学习环境和教学资源，使我有更多发展的可能性。

感谢指导老师傅沁冰老师帮助我选定毕业设计的题目和在算法设计中的耐心指导，您的专业知识和悉心教诲让我受益匪浅。感谢黄家俊师兄在论文阅读和模型理解的耐心帮助，感谢讨论班的各位师兄师姐们，让我深刻地认识到了自己的不足之处，让我学到了很多知识。

感谢广州大学 ACM 校队的教练丘凯伦、赵睿老师，因为他们校队才能越来越好。感谢与我一起并肩作战的罗泳熙和叶达之，让我大学生活更加热烈精彩，收获更多喜悦。

感谢我的舍友们，在生活中的包容和友爱，让我们共同度过了美好的时光。

感谢屈静怡同学，在我大三大四考研时期的支持与鼓励，你的陪伴让我的生活更加美好和充实。

最后感谢我的家人和朋友们在背后的支持和关心，感谢所有帮助过和关心过我的人。

## 附录

```
video = VideoReader('E:\stimulus\bg_cp_rp.avi');
width = video.width;
height = video.height;
t = 0;
sd = 1; %空间采样距离
interval = 1000/video.FrameRate; %离散帧之间的时间间隔（单位：毫秒）
tau_hp = 500; %视网膜层高通滤波的时间常数，单位为毫秒
tau_lp = 30; %信号相关性低通滤波中的时间常数，单位为毫秒
t4_lp = 30; %T4 神经元低通滤波到 LPLC2 的时间常数，单位为毫秒
t5_lp = 30; %T5 神经元低通滤波到 LPLC2 的时间常数，单位为毫秒
delay_hp = tau_hp / (tau_hp + interval); %视网膜层延迟系数
delay_hp1 = 20 / (20 + interval);
delay_lp = interval / (tau_lp + interval); %髓质层的延迟系数
delay_t4 = interval / (t4_lp + interval); %T4 神经元的延迟系数
delay_t5 = interval / (t5_lp + interval); %T5 神经元的延迟系数
time_length = 10; %用于平均的帧中的时间窗口
std_exc = 1; %兴奋场的标准差
Wexc = 3; %兴奋场中卷积核的宽度
std_inh = 2; %抑制场的标准差
Winh = 5; %抑制场中卷积核的宽度
std_cn = 5; %对比归一化场的标准差
Wcn = 11; %对比归一化场中卷积核的宽度
clip = 0.01; %半波矫正中的夹点
dc = 0.1; %半波整流中的直流分量
coe_cn = 10; %对比归一化场系数
std_w = 1; %径向空间偏差分布的标准偏差(水平)
std_h = std_w * height / width; %径向空间偏差分布的标准偏差(垂直方向)
scale_w = 3 * std_w; %尺度参数(水平)在使高斯分布的径向空间偏差
scale_h = 3 * std_h; %尺度参数(垂直方向)使高斯分布的径向空间偏置
coe_motion = 1; %求和中平行运动路径的系数
coe_contrast = 1; %求和中平行对比路径的系数
exp_ON = 0.9; %T4 神经元反应指数
exp_OFF = 0.5; %T5 神经元反应指数
LPLC2 = zeros(1,time_length); %视觉投射神经元 LPLC2 反应
LPLC2_OUT = zeros(1,video.NumFrames - 1);
number = 1;

Photoreceptor = zeros(height,width,2); %光感受器(视网膜)层
ONs = zeros(height,width,2); %ON 通道
OFFs = zeros(height,width,2); %OFF 通道
```

```

ONs_Compressed = zeros(height,width,2);    %ON 通道压缩信号
OFFs_Compressed = zeros(height,width,2);    %OFF 通道压缩信号
ONs_delay = zeros(height,width,2);          %ON 通道延迟信号
OFFs_delay = zeros(height,width,2);          %OFF 通道延迟信号
ONs_Contrast = zeros(height,width,2);        %ON 对比视觉通道
OFFs_Contrast = zeros(height,width,2);        %OFF 对比视觉通道
T4s = zeros(height,width,2,4);              %T4 髓质神经元-维宽*高* 2 * 4(基本方向), 1->
向右, 2->向左, 3->向下, 4->向上
T5s = zeros(height,width,2,4);              %小叶 T5 神经元-维宽*高* 2 * 4(基本方向), 1->
向右, 2->向左, 3->向下, 4->向上
LPLC2_mag = zeros(height,width,2);          %LPLC2 运动强度
LPLC2_dir = zeros(height,width);            %LPLC2 运动的方向
View_Centroid = [fix(height/2),fix(width/2)];%质心
%View_Center = [0,0];
c_on_pre = zeros(height,width);
c_off_pre = zeros(height,width);
mdd = zeros(height,width);
photos = zeros(height,width);
Radial_Bias = MakeRadialBiasDistribution(std_h, std_w, width, height, scale_w, scale_h);
GB_ExcKernel = MakeGaussianKernel(std_exc,Wexc);
GB_InhKernel = MakeGaussianKernel(std_inh,Winh);
Kernel_Constrast = MakeGaussianKernel(std_cn, Wcn);
maxMotion_off = 0;
maxMotion_on = 0;
view_on_x=0;
view_on_y=0;
view_off_x=0;
view_off_y=0;
mr = zeros(height,width);
ml = zeros(height,width);
md = zeros(height,width);
mu = zeros(height,width);
img = read(video, 30);
for num = 2:video.NumFrames
    maxMotion_off = 0;
    maxMotion_on = 0;
    t = t + 1;
    img1 = read(video, num - 1);
    img2 = read(video, num);
    img1=double(rgb2gray(img1));
    img2=double(rgb2gray(img2));
    LPLC2_Region = zeros(1,4);                %LPLC2 视觉投射神经元全群体区域反应-维
数 4 * 1 向量, 1->左上, 2->右上, 3->左下, 4->右下
    cur_t = mod(t,2) + 1;

```

```

pre_t = mod(t-1,2) + 1;
cur_avg = mod(t - 1,time_length) + 1;
maxMotion = 0;
tmpSum = 0;
%视网膜层处理
Photoreceptor(:, :, cur_t) = Highpass(img1, img2, Photoreceptor(:, :, pre_t), delay_hp);
%Photoreceptor(:, :, cur_t) = img2 - img1;

%高斯模糊
LA = Gaussian_Filter(Photoreceptor, Wexc, GB_ExcKernel, Winh, GB_InhKernel, cur_t,
width, height);

%半波整流
for y = 1:height
    for x = 1:width
        ONs(y, x, cur_t) = Halfwave_ON(LA(y, x), ONs(y, x, pre_t), clip, dc);
        OFFs(y, x, cur_t) = Halfwave_OFF(LA(y, x), OFFs(y, x, pre_t), clip, dc);
    end
end
%信号压缩
for y = 1:height
    for x = 1:width
        ONs_Compressed(y, x, cur_t) =
normaliseContrast(Kernel_Constrast, ONs, y, x, cur_t, width, height, coe_cn);
        OFFs_Compressed(y, x, cur_t) =
normaliseContrast(Kernel_Constrast, OFFs, y, x, cur_t, width, height, coe_cn);
    end
end
ONs_delay(:, :, cur_t) = Lowpass(ONs_Compressed(:, :, cur_t), ONs_delay(:, :, pre_t),
delay_lp);
OFFs_delay(:, :, cur_t) = Lowpass(OFFs_Compressed(:, :, cur_t), OFFs_delay(:, :, pre_t),
delay_lp);

%对比视觉通路和运动相关通路
for y = sd + 1:height - sd
    for x = sd + 1:width - sd

        %运动相关
        %向右运动
        T4s(y, x, cur_t, 1) =
EMD(ONs_Compressed(y, x, cur_t), ONs_Compressed(y, x + sd, cur_t), ONs_delay(y, x, cur_t), ONs_d
elay(y, x + sd, cur_t));
        T5s(y, x, cur_t, 1) =
EMD(OFFs_Compressed(y, x, cur_t), OFFs_Compressed(y, x + sd, cur_t), OFFs_delay(y, x, cur_t), OFF

```

```

s_delay(y,x+sd,cur_t));
    %向左运动
    T4s(y,x,cur_t,2) =
EMD(ONs_Compressed(y,x,cur_t),ONs_Compressed(y,x-sd,cur_t),ONs_delay(y,x,cur_t),ONs_de
lay(y,x-sd,cur_t));
    T5s(y,x,cur_t,2) =
EMD(OFFs_Compressed(y,x,cur_t),OFFs_Compressed(y,x-sd,cur_t),OFFs_delay(y,x,cur_t),OFFs
_delay(y,x-sd,cur_t));
    %向下运动
    T4s(y,x,cur_t,3) =
EMD(ONs_Compressed(y,x,cur_t),ONs_Compressed(y+sd,x,cur_t),ONs_delay(y,x,cur_t),ONs_d
elay(y+sd,x,cur_t));
    T5s(y,x,cur_t,3) =
EMD(OFFs_Compressed(y,x,cur_t),OFFs_Compressed(y+sd,x,cur_t),OFFs_delay(y,x,cur_t),OFF
s_delay(y+sd,x,cur_t));
    %向上运动
    T4s(y,x,cur_t,4) =
EMD(ONs_Compressed(y,x,cur_t),ONs_Compressed(y-sd,x,cur_t),ONs_delay(y,x,cur_t),ONs_de
lay(y-sd,x,cur_t));
    T5s(y,x,cur_t,4) =
EMD(OFFs_Compressed(y,x,cur_t),OFFs_Compressed(y-sd,x,cur_t),OFFs_delay(y,x,cur_t),OFFs
_delay(y-sd,x,cur_t));

    LPLC2_mag(y,x,1) =
sqrt((T4s(y,x,cur_t,1)-T4s(y,x,cur_t,2))^2+(T4s(y,x,cur_t,3)-T4s(y,x,cur_t,4))^2);%on
    LPLC2_mag(y,x,2) =
sqrt((T5s(y,x,cur_t,1)-T5s(y,x,cur_t,2))^2+(T5s(y,x,cur_t,3)-T5s(y,x,cur_t,4))^2);%off

    %T4 和 T5 神经元低通
    T4s(y,x,cur_t,1) = Lowpass(T4s(y,x,cur_t,1),T4s(y,x,pre_t,1),delay_t4);
    T4s(y,x,cur_t,2) = Lowpass(T4s(y,x,cur_t,2),T4s(y,x,pre_t,2),delay_t4);
    T4s(y,x,cur_t,3) = Lowpass(T4s(y,x,cur_t,3),T4s(y,x,pre_t,3),delay_t4);
    T4s(y,x,cur_t,4) = Lowpass(T4s(y,x,cur_t,4),T4s(y,x,pre_t,4),delay_t4);
    T5s(y,x,cur_t,1) = Lowpass(T5s(y,x,cur_t,1),T5s(y,x,pre_t,1),delay_t5);
    T5s(y,x,cur_t,2) = Lowpass(T5s(y,x,cur_t,2),T5s(y,x,pre_t,2),delay_t5);
    T5s(y,x,cur_t,3) = Lowpass(T5s(y,x,cur_t,3),T5s(y,x,pre_t,3),delay_t5);
    T5s(y,x,cur_t,4) = Lowpass(T5s(y,x,cur_t,4),T5s(y,x,pre_t,4),delay_t5);
    %局部运动方向和强度
    mr(y,x) = PolarityChannelSummation(ReLU(T4s(y,x,cur_t,1)*coe_motion),
ReLU(T5s(y,x,cur_t,1)*coe_motion), exp_ON, exp_OFF);
    ml(y,x) = PolarityChannelSummation(ReLU(T4s(y,x,cur_t,2)*coe_motion),
ReLU(T5s(y,x,cur_t,2)*coe_motion), exp_ON, exp_OFF);
    md(y,x) = PolarityChannelSummation(ReLU(T4s(y,x,cur_t,3)*coe_motion),
ReLU(T5s(y,x,cur_t,3)*coe_motion), exp_ON, exp_OFF);

```

```

        mu(y,x) = PolarityChannelSummation(ReLU(T4s(y,x,cur_t,4)*coe_motion),
ReLU(T5s(y,x,cur_t,4)*coe_motion), exp_ON, exp_OFF);

        if maxMotion_on < LPLC2_mag(y,x,1)
            maxMotion_on = LPLC2_mag(y,x,1);
            view_on_x=x;view_on_y=y;
        end
        if maxMotion_off< LPLC2_mag(y,x,2)
            maxMotion_off = LPLC2_mag(y,x,2);
            view_off_x=x;view_off_y=y;
        end
    end
end
if t==1
    View_Center(2) = view_off_x;
    View_Center(1) = view_off_y;
else
    if abs(view_off_x-View_Center(2))>50||abs(view_off_y-View_Center(1))>50
        View_Center(2) = view_off_x;
        View_Center(1) = view_off_y;
    end
end
if View_Center(1)==0
    continue
end
for y = max(View_Center(1)-50,1):min(View_Center(1)+50,height)
    for x = max(View_Center(2)-50,1):min(View_Center(2)+50,width)
        if y <= View_Center(1) && x <= View_Center(2)
            LPLC2_Region(1) = LPLC2_Region(1) + GELU_Precise(ml(y,x) - mr(y,x));
            LPLC2_Region(1) = LPLC2_Region(1) + GELU_Precise(mu(y,x) - md(y,x));
        else
            if y <= View_Center(1) && x >= View_Center(2)
                LPLC2_Region(2) = LPLC2_Region(2) + GELU_Precise(mr(y,x) -
ml(y,x));
                LPLC2_Region(2) = LPLC2_Region(2) + GELU_Precise(mu(y,x) -
md(y,x));
            else
                if y >= View_Center(1) && x <= View_Center(2)
                    LPLC2_Region(3) = LPLC2_Region(3) + GELU_Precise(ml(y,x) -
mr(y,x));
                    LPLC2_Region(3) = LPLC2_Region(3) + GELU_Precise(md(y,x) -
mu(y,x));
                else
                    if y >= View_Center(1) && x >= View_Center(2)

```

```

                                LPLC2_Region(4) = LPLC2_Region(4) +
GELU_Precise(mr(y,x) - ml(y,x));
                                LPLC2_Region(4) = LPLC2_Region(4) +
GELU_Precise(md(y,x) - mu(y,x));
                                end
                                end
                                end
                                end
                                end
                                end
                                end

LPLC2(cur_avg) = Leaky_ReLU(LPLC2_Region(1)) * Leaky_ReLU(LPLC2_Region(2)) *
Leaky_ReLU(LPLC2_Region(3)) * Leaky_ReLU(LPLC2_Region(4));

if t >= time_length
    LPLC2_OUT(t-time_length+1) = mean(LPLC2);
    number = mod(number,time_length) + 1;
end
end
for i = t-time_length+2:video.NumFrames - 1
    LPLC2_OUT(i) = mean(LPLC2(number:end));
    number = mod(number,time_length) + 1;
end
LPLC2_OUT = (LPLC2_OUT - min(LPLC2_OUT)) / (max(LPLC2_OUT) - min(LPLC2_OUT));
plot(1:t,LPLC2_OUT)

```