基于注意力和长短时记忆网络的视觉里程计

阮晓钢[1,2], 余鹏程[1,2], 朱晓庆[1,2]
(1. 北京工业大学信息学部, 北京 100124; 2. 计算智能与智能系统北京市重点实验室, 北京 100124)

摘 要: 近年来通过利用视觉信息估计相机的位姿,实现对无人车的定位成为研究热点,视觉里程计是其中的重要组成部分. 传统的视觉里程计需要复杂的流程如特征提取、特征匹配、后端优化,难以求解出最优情况. 因此,提出融合注意力和长短时记忆网络的视觉里程计,通过注意力机制增强的卷积网络从帧间变化中提取运动特征,然后使用长短时记忆网络进行时序建模,输入RGB图片序列,模型端到端地输出位姿. 在公开的无人驾驶KITTI数据集上完成实验,并与其他算法进行对比. 结果表明,该方法在位姿估计上的误差低于其他单目算法,定性分析显示该算法具有较好的泛化能力.

关键词: 深度学习; 注意力机制; 时序建模; 视觉里程计; 位姿估计; 镜像网络
中图分类号: TP 242.6 **文献标志码**: A **文章编号**: 0254-0037(2021)08-0815-09
doi: 10.11936/bjutxb2021010015

Visual Odometer Based on Attention and LSTM

RUAN Xiaogang[1,2], YU Pengcheng[1,2], ZHU Xiaoqing[1,2]
(1. Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China;
2. Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing 100124, China)

Abstract: In recent years, the use of visual information to estimate the pose of the camera to realize the positioning of unmanned vehicles has become a research hotspot. Visual odometry is an important part of it. Traditional visual odometry requires complex processes such as feature extraction, feature matching, and post-processing. It is difficult to solve the optimal situation. Therefore, a visual odometer that combines attention and long short-term memory (LSTM) was proposed in this paper. The convolutional network was enhanced by the attention mechanism, which extracted motion features from the changes between frames. Then, the long and short-term memory network was used for timing modeling. The input was a sequence of RGB pictures, and a pose of end-to-end was output by the model. The experiment was completed on the public unmanned driving KITTI data set and compared with other algorithms. Results show that the error of the method in pose estimation is lower than that of other monocular algorithms, and through qualitative analysis, it has good generalization ability.

Key words: deep learning; attention mechanism; sequence modeling; visual odometry; pose estimation; symmetric network

视觉里程计[1](visual odometry, VO)解决同时定位和地图构建中的定位问题,主要用来求解相机在未知环境中的位姿并绘制出相机运动的轨迹. 该技术被广泛应用于无人驾驶、机器人、潜航器等系

收稿日期: 2020-01-05
基金项目: 国家自然科学基金资助项目(61773027); 北京市自然科学基金资助项目(4202005)
作者简介: 阮晓钢(1958—), 男, 教授, 主要从事机器人、自动控制与人工智能方面的研究, E-mail: adrxg@bjut.edu.cn
通信作者: 朱晓庆(1987—), 男, 讲师, 主要从事机器人、机器学习方面的研究, E-mail: alex.zhuxq@gmail.com

统,是继惯性导航、车轮里程计等定位技术之后的一种新的导航技术. 视觉里程计利用相机捕获连续运动时间内的图片序列,然后通过算法计算出相邻图片间的运动关系,最终输出相机的相对姿态,其中包括旋转和平移信息,这一过程也被称为视觉位姿估计[2].

视觉里程计的早期研究是针对火星探索计划进行的,2004年Nister等提出并搭建了最早的VO系统,为后续的VO系统的研究提供了优秀的范例和参考. 视觉里程计可以分为单目、双目和深度(RGB-D)相机视觉里程计. 根据采用的求解方法又可以划分为基于间接法[3-4]和直接法[5-6]的视觉里程计. 间接方法在连续帧的匹配特征之间执行几何变换来计算相机运动的大致位姿,然后通过局部或全局的光束平差法(bundle adjustment, BA)进行位姿优化. 其中基于特征法的位姿估计充分利用了图片的几何信息,主要包含特征提取、特征匹配、误匹配剔除和运动估计等步骤. 而基于直接法的位姿估计使用了图片的光度信息,直接通过计算像素值来估计相机的运动. 但这些系统都存在一定的问题,直接法是基于相邻图片之间的灰度不变假设,然后再求解位姿,这种假设在面对光照变化明显的场景时难以成立,而特征点法只适用于具有明显特征的场景,当面对纹理单一、存在动态物体的场景时,难以提取有效的特征.

深度学习(deep learning)被广泛应用于目标检测、目标追踪等诸多计算机视觉领域[7-8]. 将其应用到同步定位与建图领域(simultaneous location and mapping, SLAM)也是目前国内外研究的趋势. Roberts等[9]将每个图片帧划分为网格单元并计算每个单元的光流,然后使用K近邻(K-nearest neighbors,KNN)算法来估计当前位姿的变化. 随后一些研究人员提出通过卷积神经网络(conventional neural network,CNN)从光流图片序列中估计相机运动位姿[10],例如PCNN VO[11]、Flow-odometry[12]、LS-V[13]等. CNN可以自动对图片进行不同尺度的特征提取,省去了传统方法中烦琐的特征提取过程. 但是,由于VO需要考虑连续图片序列之间的相关信息,需要处理和发现图片之间更多的低层几何变换信息,而长短时记忆(long short-term memory,LSTM)网络能实现这种时序上的关联. 综上所述,本文放弃传统复杂且基于一定条件的系统设计,采用深度学习构建端到端的模型,提出了一种融合注意力、卷积和长短时记忆网络的视觉里程计ALC-VO(attention LSTM CNN-visual odometry). 注意力机制能提高系统的特征提取能力,在CNN基础上增加通道注意力和空间注意力处理图片,学习图片中的运动特征,而不是具体的语义信息. 而且视觉里程计本身是一种时序上对相机的位姿估计,通过LSTM自动学习图片间的关联,能够利用历史信息完成对当前位姿的有效估计. 为了充分利用已有的数据,将图片进行正序的训练,还将图片的逆序投入到神经网络中进行训练.

在公开数据集KITTI上进行实验,结果证明本文方法在位姿估计精度上优于传统的单目视觉里程计算法. 论文的具体结构如下:第1部分主要介绍本文所提出的模型. 第2部分给出了模型在无人驾驶KITTI数据集下的实验结果与分析. 第3部分得出结论,并对下一阶段的工作进行展望.

1 模型设计

本文构建的视觉里程计模型ALC-VO的详细结构如图1所示,ALC-VO结合CNN层、ATTENTION层、LSTM层,以图片序列为输入,首先使用融合了注意力机制的CNN提取相邻图片间的局部特征,然后利用LSTM时序建模,最后通过全连接层输出相机的相对位姿信息. 同时采用图2的镜像网络,对模型进行正向和逆向的训练,正向和逆向的区别在于图片输入的顺序是否相反,在正常测试及验证模型的时候则不需要镜像网络.

1.1 基于CNN的运动特征提取

目前已经有很多网络能获取图片的语义信息,如VGG Net、Google Net等,然而它们主要用于分类或者目标检测等相关任务. 而相机位姿估计与图片分类任务区别很大,首先图片分类每次只需要提取一张图片的特征,而视觉里程计是通过2张图片来计算出相机的位姿,更依赖图片之间的几何特征信息,属于深度学习中的回归问题. 因此,采用诸如VGG Net结构来解决视觉里程计问题并不是最优选择. 而光流可以表示相邻时刻图片间的运动关系,所以本文参照了光流神经网络模型Flow-Net,通过修改Flow-Net的子网络FlowNetSimple构建了ALC-VO中的CNN部分,用于提取图片运动特征.

网络的输入为相邻2帧的图片,2张图片均为数据集中的原始RGB图片,为了适应本文所提出的网络,将图片大小修改为1 280×384,并将2张图片进行第3维度上的串联,组成第3维度为6的数据

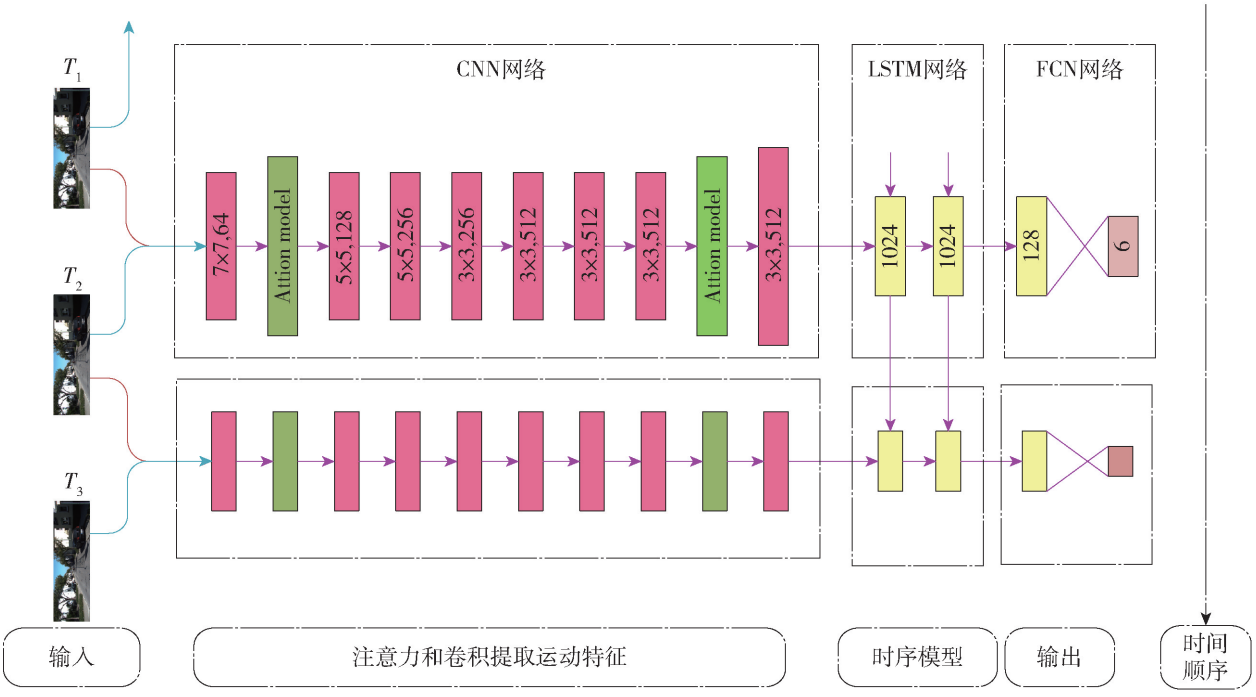


图 1　基于 ALC-VO 的视觉里程计模型

Fig. 1　Visual odometry based on ALC-VO

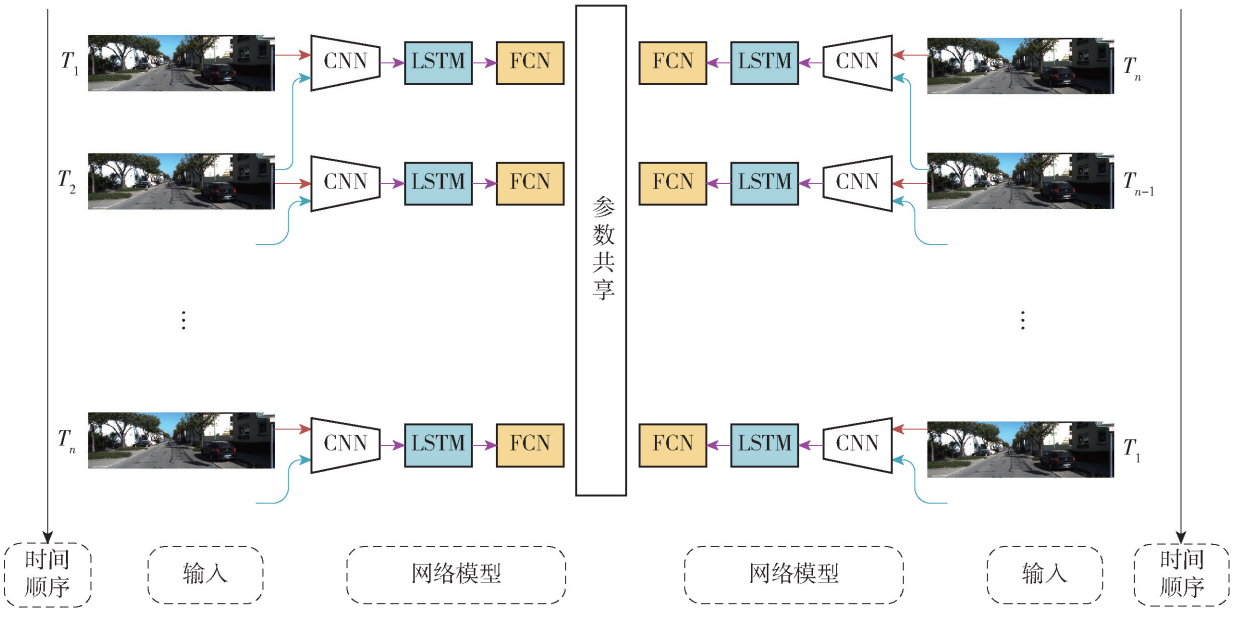


图 2　基础框架

Fig. 2　Basic framework

并输入到网络中. 逐层学习图片运动特征的过程中,在每个卷积层之后添加批量归一化(batch normalization)[14],使得卷积变换前后数据分布保持不变. 完成相邻图片间的运动特征学习后,将最后一个卷积层输出的特征图输入到后面的 LSTM. CNN 的各层参数如表 1 所示. 该 CNN 模型总共包含 8 个卷积层,其中第 1 个卷积层的卷积核大小为 7×7,第 2~4 层的卷积核大小减小为 5×5,后面 4 层的卷积核大小再次缩小到 3×3.

1.2　基于 ConvLSTM 的时序建模

LSTM 网络十分适合处理时序数据问题,可以在计算下一时刻相邻图片间的运动关系时,提供之前时刻保留的位姿信息. 因为是图片数据,所以使用了卷积长短时记忆网络 ConvLSTM[15].

表 1 CNN 的各层参数

Table 1 Parameters of each layer of CNN

网络层	卷积核	步长	填充	通道数量
Conv_1	7×7	2	3	64
Conv_2	5×5	2	3	128
Conv_3	5×5	2	2	256
Conv_4	5×5	2	2	256
Conv_5	3×3	1	1	512
Conv_6	3×3	2	1	512
Conv_7	3×3	1	1	512
Conv_8	3×3	2	1	512

ConvLSTM 在 LSTM 结构的基础上进行了改进,将权重与输入层的全连接方式改为局部连接,通过卷积运算可以更好地学习图片的空间特征.

假设当前时刻为 t,输入为 X_t 和上一时刻的隐藏状态 H_{t-1},则 ConvLSTM 输出及状态更新的计算式[15]为

$$\begin{aligned} i_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i) \\ f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \circ C_{t-1} + b_f) \\ C_t &= f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \\ o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \circ C_t + b_o) \\ H_t &= o_t \circ \tanh C_t \end{aligned} \tag{1}$$

式中:σ 为 sigmoid 激活函数; $*$ 为卷积运算; $\circ$ 为 2 个矩阵或向量对应元素相乘,称为 Hadamard 乘积; i_t、f_t、C_t、o_t、H_t 均为三维张量,其中 1 个维度是时间,其余 2 个分别是图像的长、宽.

ALC-VO 中 LSTM 结构由 2 个 ConvLSTM 叠加构成,每个 ConvLSTM 层包含 1 024 个隐藏单元,卷积核大小为 3×3. 其中第 1 个 ConvLSTM 层连接在卷积层 conv_8 之后,第 2 个 ConvLSTM 层连接在第 1 个 ConvLSTM 层之后,其输入为第 1 个 ConvLSTM 层的隐藏状态 h_t. 为了保持原始的数据分布,2 个 ConvLSTM 层中使用的激活函数为 ReLU. ALC-VO 在第 2 个 ConvLSTM 层之后添加了 2 个全连接层,神经元个数分别为 128 和 6,其中最后 1 个全连接层的输出为 ALC-VO 估计的相机相对位姿.

1.3 注意力机制

注意力模块可嵌入到卷积神经网络中进行一种简单而又有效的注意力机制部署. 其主要包含通道注意力模块及空间注意力模块. 通过使用注意力机制来增强网络架构的表达力,进一步表征出图片之间的几何关系变换,使得网络能够更加智能地学习到更重要的特征同时关注那些特征区域,并且减少学习一些不重要的特征,这也是注意力机制的本质所在,并将基于卷积的注意力模块集成到 CNN 网络架构中进行端到端的训练. 其基本结构如图 3 所示. 主要对特征进行 2 个操作,公式为

$$\begin{aligned} F' &= \sigma(\mathrm{MLP}(\mathrm{AP}(F)) + \mathrm{MLP}(\mathrm{MP}(F))) \odot F \\ F'' &= \sigma(f^{7\times7}[\mathrm{AP}(F'), \mathrm{MP}(F')]) \odot F' \end{aligned} \tag{2}$$

式中:$\odot$ 为元素级乘法; $F \in \mathbb{R}^{C\times H\times W}$ 为特征图; $F' \in \mathbb{R}^{C\times1\times1}$ 为注意力模块推断出的一维通道域注意力特征; $F'' \in \mathbb{R}^{C\times1\times1}$ 为注意力模块推断出的二维空间域注意力特征; F' 为中间输出; F'' 为最终的优化结果; σ 为 Sigmoid 函数; $f^{7\times7}$ 表示 7×7 的卷积层; AP 为平均池化; MP 为最大池化; MLP 为一个全连接层.

1.4 损失函数及优化

可以把视觉里程计估计问题看成一个条件概率问题,给定一个序列的 $n+1$ 张图片:

$$X = (X_1, X_2, \cdots, X_{n+1}) \tag{3}$$

计算得到该序列中两两相邻的图片之间的姿态:

$$Y = (Y_1, Y_2, \cdots, Y_n) \tag{4}$$

VO 估计问题看成一个条件概率问题,在给定图片序列的情况下,计算位姿的概率表示为

$$p(Y|X) = p(Y_1, Y_2, \cdots, Y_n | X_1, X_2, \cdots, X_{n+1}) \tag{5}$$

这里要解决的问题就是求解最优的网络参数 w^* 使得式(5)中的概率最大化. 式(5)表示在给定

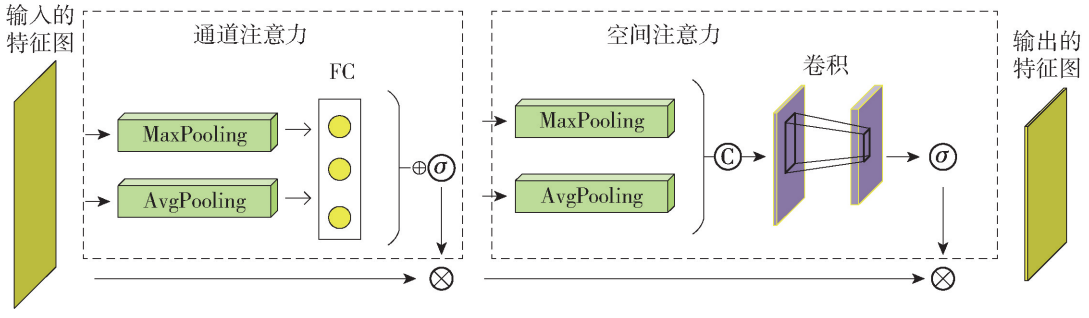


图 3 注意力模型

Fig. 3 Attention model

图片序列($Y_1, Y_2, \cdots, Y_n$)的情况下，估计相机的位置和姿态($X_1, X_2, \cdots, X_{n+1}$)是对整个视觉里程计任务的数学描述. 而在公式

$$w^* = \arg\max_w p(X|Y;w) \tag{6}$$

中增加网络参数 w，是用神经网络处理视觉里程计这个任务的数学描述. 在给定图片序列($Y_1, Y_2, \cdots, Y_n$)情况下，不同网络参数来估计位姿($X_1, X_2, \cdots, X_{n+1}$)会有不同的概率值. 优化算法需要寻找使其估计概率最大这一情况下的网络参数值，这代表整个网络达到最优状态.

图 2 中左半部分和右半部的网络结构完全对称，图片序列按照时间顺序依次经过 CNN 层、LSTM 层以及 FCN 层，输出相邻两帧之间的相对姿态. 右半部分和左半部分完全对称，只是图片序列采用逆序输入的方式，得到的输出也代表相邻两帧图片之间的相对姿态，不过是上一帧相对于当前帧的相对姿态. 误差为所有输出姿态的均方误差，表示为

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N} \| P_{1i} - \hat{P}_{1i} \|_2^2 + \beta_1 \| \Phi_{1i} - \hat{\Phi}_{1i} \|_2^2 + \| P_{2j} - \hat{P}_{2j} \|_2^2 + \beta_2 \| \Phi_{2j} - \hat{\Phi}_{2j} \|_2^2 \tag{7}$$

式中：$\hat{P}_{1i}$、$\hat{\Phi}_{1i}$分别为第 i 对样本按正序输入的位移和转角；$\hat{P}_{2j}$、$\hat{\Phi}_{2j}$分别为第 j 对样本按逆序输入的位移和转角；P_{1i}、Φ_{1i}、P_{2j}、Φ_{2j}分别为 Ground Truth 的位移和转角. β_1、β_2 分别为正序输入和逆序输入的转角误差尺度因子.

2　实验结果

本实验采用显卡 Nvidia GeForce 2080ti 来训练和测试模型，CPU 为 Intel 至强 E5-2673-V3. 在深度学习框架 Pytorch 下进行相关算法的设计. 使用 Adam(adaptive moment estimation)优化器进行 100 个 epoch 的训练，并将学习率设置为 10×10^{-2}，同时引入 Dropout 和 Early Stopping 技术来防止模型过拟合. 神经网络输入预处理图片的大小为 1 280 × 384，训练时采用 2 块上述 GPU 进行同时训练，训练 1 个 epoch 需要 0.1 h 左右.

2.1　实验数据集

KITTI Visual Odometry[16]是 Geiger 等开源的汽车驾驶数据集. 可广泛用于评估各种 VO 或者 SLAM 算法. KITTI 数据集前 11 个序列信息如表 2 所示.

KITTI VO benchmark 共包含有 22 个场景图片，每个场景都包含有双目摄像机拍摄的一系列的图片. 不过本文只使用双目数据集中的单目图片. 其

表 2　KITTI 数据集中序列 00-10 信息

Table 2　Information of sequences 00-10 in KITTI

序列	图片总数	图片尺寸	总长度/m
00	4 541	1 241 × 376	3 724
01	1 101	1 241 × 376	2 453
02	4 661	1 241 × 376	5 067
03	801	1 241 × 376	560
04	271	1 226 × 370	393
05	2 761	1 226 × 370	2 205
06	1 101	1 226 × 370	1 232
07	1 101	1 226 × 370	694
08	4 071	1 226 × 370	3 222
09	1 591	1 226 × 370	1 705
10	1 201	1 226 × 370	919

中序列 00—03 的图片尺寸为 1 241 × 376，序列 04—10 的图片尺寸 1 226 × 370，为了符合网络对输入数据的要求，将所有图片的尺寸调整为 1 280 × 384. 22 个序列中只有前 11 个序列提供了每个图片对应的真实姿态数据，部分场景中含有动态移动的物体以及明暗的显著变换，部分场景中汽车行驶速度高达 90 km/h. 总体信息如表 2 所示.

考虑到训练集和验证集的图片种类划分，本文使用 08 以前的序列对 ALC-VO 模型进行训练，并选择一定数量的数据进行验证，然后使用 08、09、10 序列对 ALC-VO 模型进行测试. 训练集和验证集的所有图片均由双目相机中的左相机所采集.

2.2　训练过程

首先，将讨论影响模型的超参数学习率 η，Loss 曲线如图 4 所示. 本实验中 η 被设置为[0.40, 0.20, 0.10, 0.05, 0.01]，Loss 曲线刚开始下降得很快，后面就是震荡和随机抖动；最终使用 $\eta = 0.40$ Loss 曲线震荡得非常明显，这样的参数没办法使得网络获得较好的性能. 同样如果 η 值太小(如 0.01)，那么收敛速度可能会太慢，需要更长的时间来得到较好的网络参数.

2.3　误差分析

将训练好的 ALC-VO 模型在测试集序列 08、09、10 上进行测试，并与其他先进的 VO 系统进行比较，通过 KITTI VO/SLAM 官方的评价方式进行评测：路径长度为 100 ~ 800 m，而且汽车速度不相同(不同场景中汽车的速度不相同)，评价指标为

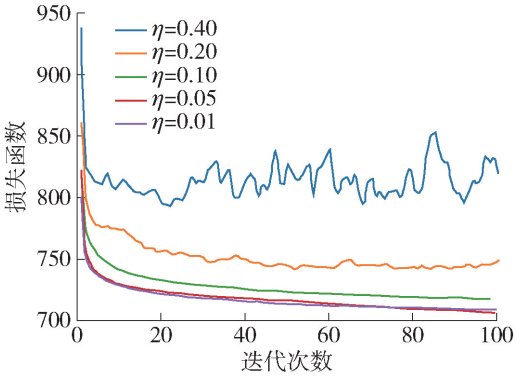


图 4 不同学习率下的 Loss 变化

Fig. 4 Loss with different learning rates

平移误差和旋转误差的均方根误差(RMSE). 在误差中 t 为平移误差百分比,r 为旋转误差,分别表示为

$$t = \frac{s_t - s_p}{s_t} \tag{8}$$

$$r = \frac{R_{error}}{s_t} \tag{9}$$

式中:s_t 为地面真实的轨迹总长度;s_p 为里程计预测的轨迹总长度;R_{error} 为总的旋转误差.

不同算法在测试集上的具体表现如表 3 所示,表中各序列的位姿定量评价指标为不同路径长度的平均平移误差和平均旋转误差. 实验结果表明,本文算法要优于 PCNN-VO、VISO2-Mono[17]、SVR-VO[18]、FLOW-VO、Zhou 等[19]、Deep-VO-Feat[20]、CC[21]、SC-SFM Learner[22]算法,证明了本文模型的可行性. 基于光流的 FLOW-VO 也有良好的表现,这表明在运动估计中,基于神经网络的光流方法可行,用光流来表征图片的运动特征是一种较优的选择.

VISO2-S-VO 与 VISO2-M-VO 均为采用传统方式求解 VO 的算法,两者不同之处在于 VISO2-M-VO 为单目算法,VISO2-S-VO 为双目算法. 为了对比传统方法的实验效果,将 VISO2-S-VO 与 VISO2-M-VO 在 KITTI 数据集的场景中进行测试. 如图 5 所示,即为本文方法所估计的 VO 在不同路径长度和速度下的平移和旋转角度的平均 RMSE. 可以看到本文所提出的方法比 VISO2-M 单目 VO 的效果要好,但是比 VISO2-S-VO 的定位精度要低. 从整体来看,除了双目的 VISO2-S-VO,本文的模型都优于其余的单目算法. 但随着相机运动速度的增加,ALC-VO 的误差有增大的趋势,原因主要是训练集中缺少运动速度较快的样本,因此网络在预测高速运动下平移和旋转信息时相对来说误差偏大.

表 3 各方法平移和旋转误差

Table 3 Translation and rotation errors of each method

序列	ALC-VO		文献[19]		Deep-VO-Feat		CC		SC-SFM Learner	
	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})
08	5.91	0.015 0	13.42	0.042 0	9.15	0.036 4	8.46	0.038 1	8.55	0.035 4
09	9.12	0.035 0	11.34	0.040 8	9.07	0.038 0	7.71	0.023 2	7.60	0.021 9
10	7.88	0.034 3	15.26	0.041 2	9.60	0.034 1	9.87	0.044 7	10.77	0.046 3
平均值	7.63	0.028 1	13.34	0.041 3	9.27	0.036 1	8.68	0.035 3	8.97	0.034 5
序列	VISO2-S-VO		VISO2-M-VO		PCNN-VO		FLOW-VO		SVR-VO	
	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})	t/%	r/((°)·m^{-1})
08	5.65	0.012 7	19.39	0.039 3	7.60	0.018 7	9.98	0.054 4	14.44	0.030 0
09	8.80	0.033 6	9.26	0.027 9	6.75	0.025 2	12.64	0.080 4	8.70	0.026 6
10	7.56	0.032 1	27.55	0.040 9	21.23	0.040 5	11.65	0.072 8	18.81	0.026 5
平均值	7.33	0.026 1	18.73	0.036 6	11.86	0.028 2	11.42	0.069 2	13.98	0.028 0

2.4 轨迹可视化

为了更好地进行定性分析,将 ALC-VO、VISO2-S-VO、VISO2-M-VO、PCNN-VO、FLOW-VO 、SVR-VO 在序列 09、10 上估计的运动轨迹进行了可视化处理,如图 6 所示. 图中由真实位姿数据产生的轨迹作为测量标准. 可以看出 5 种算法大致上都恢复出了真实运动轨迹的形状,而 ALC-VO 模型估计的轨迹要优于 VISO2-M-VO、PCNN-VO、FLOW-VO、SVR-VO,更接近于真实轨迹. 但是 ALC-VO 的精度要略低于 VISO2-S-VO. 原因一,VISO2-S-VO 是双目算

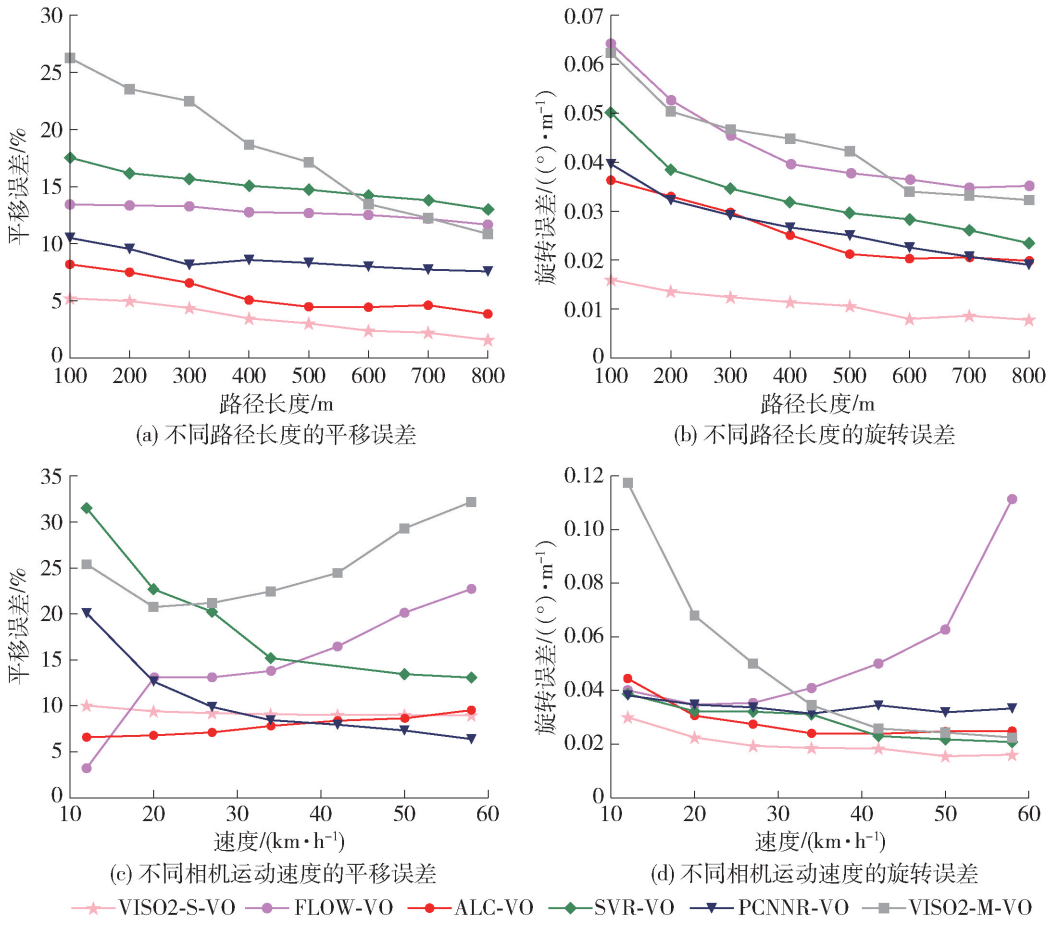


图 5　不同算法在 4 种指标下的误差

Fig. 5　Errors of different algorithms based on four indices

法，VISO2-S-VO 可以通过左目相机和右目相机拍摄的 2 幅图片获取尺度信息，从而位姿估计更加准确；原因二，ALC-VO 的训练样本不够多，没有达到网络最优的状态.

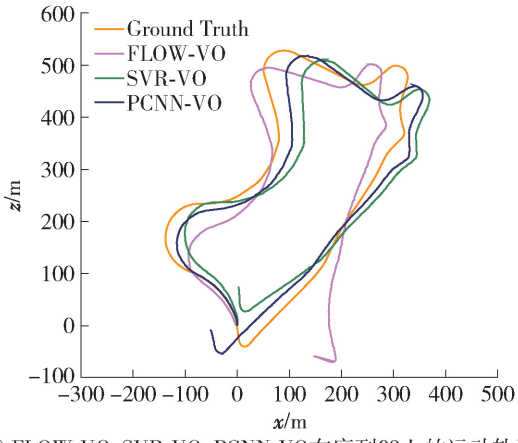
另外，本文还在序列 11、12 上对 ALC-VO 模型行了测试，序列 11、12 没有提供真实位姿数据，且相机的速度也快于前面的序列，因此相邻图片间的相机运动幅度更加大，这将十分考验算法的性能. 图 7 展示了不同算法估计的运动轨迹的可视化结果，因为缺少真实数据，所以使用 VISO2-M-VO、VISO2-S-VO 的运算结果与本文算法进行对比，并将 VISO2-S-VO 作为参考. 从图 7 中可以看出相较于 VISO2-M-VO 方法，ALC-VO 的运动轨迹更接近于 VISO2-S-VO，这展现出了本文提出的模型有较好的泛化能力.

2.5　消融实验

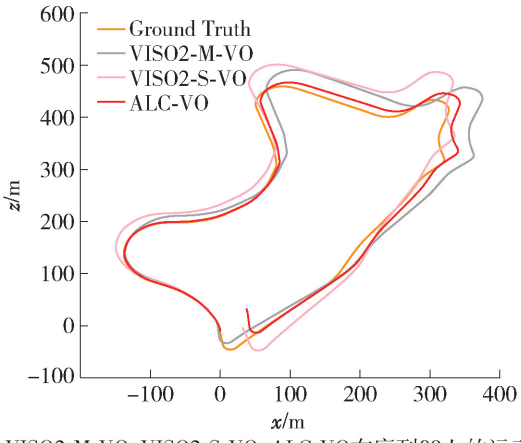
另外，为了验证注意力结构和 LSTM 网络结构对算法的提升效果，本文在 KITTI 数据集中进行了不同模块的训练以及测试. 实验结果如表 4 所示，CNN 算法的平移误差为 12.32%，旋转误差为 0.046 0°/m，将注意力结构单独嵌入到 CNN 中，平移误差降低至 8.25%，旋转误差降低至 0.035 0°/m，验证了 LSTM 结构的有效性. 将 LSTM 结构单独嵌入到 CNN 算法中，平移误差降低至 8.83%，旋转误差降低至 0.037 0°/m，验证了本文提出算法整体结构的有效性. 将 2 个结构同时嵌入到 CNN 算法中，平移误差降低至 7.63%，旋转误差降低至 0.028 1°/m，验证了本文算法的有效性.

2.6　运行速度对比

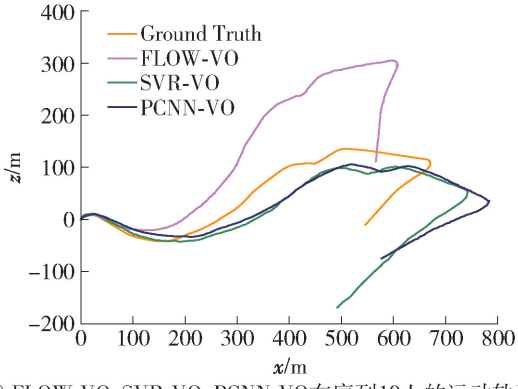
最后，检测算法所消耗的时间，本文对不同算法进行了对比，具体内容如表 5 所示. 从整体上看本文算法所消耗的时间低于 SVR-VO 和 PCNN-VO，但是高于 FLOW-VO. 在特征提取的环节本文参考了 FLOW-VO 的网络结构，同时本文模型增加注意力模块及长短时记忆网络，从而增加了算法的计算量，导致计算时间的增加. 通过定量分析，ALC-VO 较 FLOW-VO 在精度上提高了 33.2%，同时时间上增加了 25.7%.



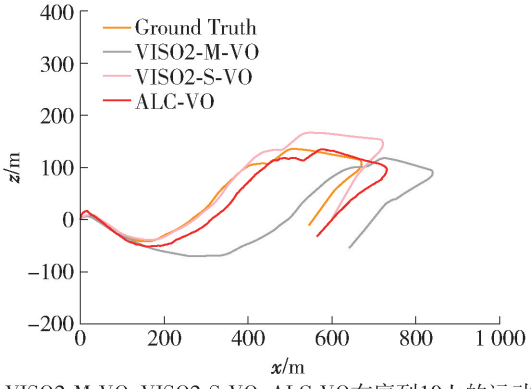
(a) FLOW-VO、SVR-VO、PCNN-VO在序列09上的运动轨迹



(b) VISO2-M-VO、VISO2-S-VO、ALC-VO在序列09上的运动轨迹



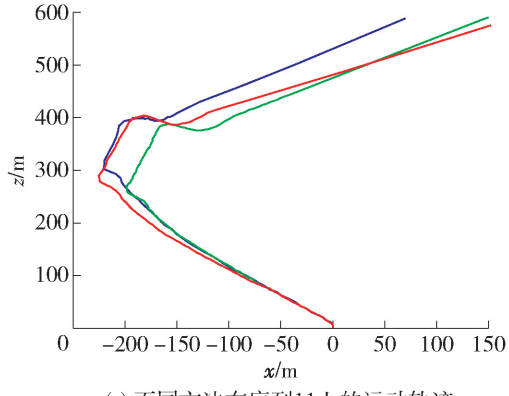
(c) FLOW-VO、SVR-VO、PCNN-VO在序列10上的运动轨迹



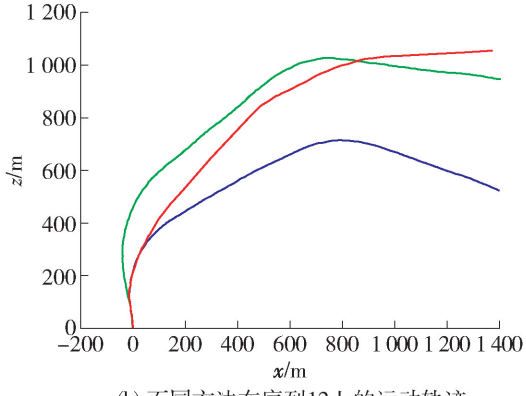
(d) VISO2-M-VO、VISO2-S-VO、ALC-VO在序列10上的运动轨迹

图 6 不同算法在序列 09、10 上生成的运动轨迹

Fig. 6 Motion trajectories generated by different algorithms on sequence 09 and 10



(a) 不同方法在序列11上的运动轨迹



(b) 不同方法在序列12上的运动轨迹

VISO2-M-VO VISO2-S-VO ALC-VO

图 7 不同算法在序列 11、12 上生成的运动轨迹

Fig. 7 Motion trajectories generated by different algorithms on sequence 11 and 12

表 4 不同网络结构的误差结果

Table 4 Error results of different network structures

序列	ALC-VO		CNN + LSTM		CNN + 注意力		CNN	
	t/%	r/(°·m^{-1})	t/%	r/(°·m^{-1})	t/%	r/(°·m^{-1})	t/%	r/(°·m^{-1})
平均值	7.63	0.028 1	8.83	0.037 0	8.25	0.035 0	12.32	0.046 0

表 5　不同方法的时间对比结果

Table 5　Time comparison results of different methods

方法	每帧执行时间/ms
SVR-VO	333
PCNN-VO	302
FLOW-VO	105
ALC-VO	132

3　结论

1）本文提出了一种基于注意力和 LSTM 时序建模的单目视觉里程计算法，通过融合 CNN、注意力、LSTM 构建整体的网络，并利用一个对称的镜像网络，使其能够学习图片之间的深层的几何运动信息.

2）所提出的算法相较于传统的视觉里程计算法，具有更好的准确性，同时摒弃了相机标定特征提取特征匹配等复杂过程，在不同场景下更易于实现，具有较高的稳定性.

3）与其他算法相比，本文算法不仅对视角和光线变化更鲁棒，而且在位姿估计误差的精度方面有显著的提升. 不足之处是，所提出的网络模型有较多参数，时间复杂度并非最优. 因此，未来的工作将通过知识蒸馏的方式减少网络参数，对本文算法的运行速度进行优化.

参考文献：

[1] SCARAMUZZA D, FRAUNDORFER F. Visual odometry: tutorial [J]. IEEE Robotics & Automation Magazine, 2011, 18(4): 80-92.

[2] 张亮，蒋荣欣，陈耀武. 移动机器人在未知环境下的同步定位与地图重建方法[J]. 控制与决策，2010，25(4)：515-520.
ZHANG L, JIANG R X, CHEN Y W. Simultaneous localization and map reconstruction of mobile robot in unknown environment [J]. Control and Decision, 2010, 25 (4): 515-520. (in Chinese)

[3] DAVISON A J, REID I D, MOLTON N D, et al. MonoSLAM: real-time single camera SLAM [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(6): 1052-1067.

[4] KLEIN G, MURRAY D. Parallel tracking and mapping for small AR workspaces[C]//Proc of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality. Piscataway: IEEE, 2007: 225-234.

[5] NEWCOMBE R A, LOVEGROVE S J, DAVISON A J. DTAM: dense tracking and mapping in real-time[C]// Proc of IEEE International Conference on Computer Vision. Piscataway: IEEE, 2011: 2320-2327.

[6] ENGEL J, SCHÖPS T, CREMERS D. LSD-SLAM: large-scale direct monocular SLAM [C]// Proc of Computer Vision-ECCV. Berlin: Springer, 2014: 834-849.

[7] KARPATHY A, TODERICI G, SHETTY S, et al. Large-scale video classification with convolutional neural networks [C]// Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2014: 1725-1732.

[8] REN S, HE K, GIRSHICK R, et al. Faster r-CNN: towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(6): 1137-1149.

[9] ROBERTS R, NGUYEN H, KRISHNAMURTHI K, et al. Memory-based learning for visua odometry [C]// 2008 IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2008: 47-52.

[10] KONDA K R, MEMISEVIC R. Learning visual odometry with a convolutional network [C]// Proceedings of International Conference on Computer Vision Theory and Applications. New York: ACM, 2015: 486-490.

[11] COSTANTE G, MANCINI M, VALIGI P A, et al. Exploring representation learning with cnns for frame-to-frame ego-motion estimation [J]. IEEE Robotics and Automation Letters, 2015, 1(1): 18-25.

[12] MULLER P, SAVAKIS A. Flow odometry: an optical flow and deep learning-based approach to visual odometry [C]//2017 IEEE Winter Conference on Applications of Computer Vision (WACV). Piscataway: IEEE, 2017: 624-631.

[13] DONAHUE J, HENDRICKS L A, GUADARRAMA S, et al. Long-term recurrent convolutional networks for visual recognition and description [C]// IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2015: 2625-2634.

[14] IOFFE S, SZEGEDY C. Batch normalization: accelerating deep network training by reducing internal covariate shift [J]. arXiv preprint arXiv, 2015: 1502.03167.

[15] SHI X, CHEN Z, WANG H, et al. Convolutional LSTM network: a machine learning approach for precipitation nowcasting [J]. arXiv preprint arXiv, 2015: 1506.04214.

（下转第 924 页）

Conference on Artificial Intelligence. Menlo Park: AAAI, 2019: 4610-4617.

[72] BELGHAZI M I, BARATIN A, RAJESHWAR S, et al. Mutual information neural estimation [C]// International Conference on Machine Learning. New York: ACM, 2018: 531-540.

[73] TSCHANNEN M, DJOLONGA J, RUBENSTEIN P K. On mutual information maximization for representation learning [J]. ArXiv Preprint ArXiv, 2019: 1907.13625.

[74] CHANG J L, MENG G F, WANG L F, et al. Deep self-evolution clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 42(4): 809-823.

[75] CHANG J L, GUO Y W, WANG L F, et al. Deep discriminative clustering analysis [J]. ArXiv Preprint ArXiv, 2019: 1905.01681.

[76] ZHAO J J, LU D H, MA K, et al. Deep image clustering with category-style representation [C]// European Conference on Computer Vision. Berlin: Springer, 2020: 54-70.

[77] TAPASWI M, LAW M T, FIDLER S. Video face clustering with unknown number of clusters [C]// Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway: IEEE, 2019: 5027-5036.

[78] WANG M, DENG W H. Deep face recognition with clustering based domain adaptation [J]. Neurocomputing, 2020, 393: 1-14.

[79] LIN W A, CHEN J C, CASTILLO C D, et al. Deep density clustering of unconstrained faces [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2018: 8128-8137.

[80] KHAN Z, YANG J. Bottom-up unsupervised image segmentation using FC-Dense u-net based deep representation clustering and multidimensional feature fusion based region merging [J]. Image and Vision Computing, 2020, 94: 103871.

[81] ZHOU L, WEI W Y F. DIC: deep image clustering for unsupervised image segmentation [J]. IEEE Access, 2020, 8: 34481-34491.

[82] SAHA S, SUDHAKARAN S, BANERJEE B, et al. Semantic guided deep unsupervised image segmentation [C]// International Conference on Image Analysis and Processing. Berlin: Springer, 2019: 499-510.

(责任编辑 张 蕾)

(上接第 823 页)

[16] GEIGER A, LENZ P, URTASUN R. Are we ready for autonomous driving? The kitti vision benchmark suite[C]// 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 3354-3361.

[17] GEIGER A, ZIEGLER J, STILLER C. Stereoscan: dense 3d reconstruction in real-time [C]// 2011 IEEE Intelligent Vehicles Symposium (IV). Piscataway: IEEE, 2011: 963-968.

[18] CIARFUGLIA T, COSTANTE G, VALIGI P, et al. Evaluation of non-geometric methods for visual odometry [J]. Robotics and Autonomous Systems, 2014, 62(12): 1717-1730.

[19] ZHOU T, BROWN M, SNAVELY N, et al. Unsupervised learning of depth and ego-motion from video [C]// IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 1851-1858.

[20] ZHAN H G, GARG R, CHAMARA S, et al. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction [C]// IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2018: 340-349.

[21] RANJAN J, JAMPANI V, BALLES L, et al. Competitive collaboration: joint unsupervised learning of depth, camera motion, optical flow and motion segmentation[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2019: 12240-12249.

[22] BIAN J W, LI Z C, WANG N Y, et al. Unsupervised scale-consistent depth and ego-motion learning from monocular video [J]. ArXiv Preprint ArXiv, 2019: 1908.10553.

(责任编辑 张 蕾)