[文章编号] 1003－4684(2023)02-0027-06

基于手部姿态识别的 MIDI 体感交互系统设计

周 祺，张 帅

（湖北工业大学 工业设计学院，湖北 武汉 430068）

[摘 要] 为摆脱 MIDI 音乐设备在交互方式上的限制，同时针对目前体感交互系统便携性差、价格昂贵等问题，将边缘计算设备与轻量级网络结合，提出了一种基于手部姿态识别的音乐体感交互系统。系统上位机依托于树莓派 4b，通过单目摄像头获取数据进而识别手部关键点，调用舵机追踪目标，并通过 nRF24L01 通信模块将动作指令发送给下位机，从而实现中远距离控制音乐演奏。通过在轻量级网络 ShuffleNet v2 中嵌入 SENet 通道注意力机制并进行网络瘦身操作，系统可在 200 ms 内完成体感控制任务，能满足用户手势控制、设备协同等需求。

[关键词] MIDI；树莓派；手势识别；ShuffleNet v2；体感交互

[中图分类号] TP 271 [文献标识码] A

人机交互领域如今把交互行为的相关问题作为评价设计资源、情感性、体验感受和美学的重要指标[1]。交互系统设计的灵感也越来越多地来自于以身体设计和空间设计为基础的体感动作，在姿态识别、手势识别等技术应用中尤其明显。但相关系统在 MIDI 音乐控制这类日常活动中并未得到广泛应用，且多数应用需要特定的辅助设备，如动捕手套、深度摄像头等。随着深度神经网络在计算机视觉领域研究的深入，平面图像的特征提取和手势识别的能力越来越强，使用廉价易用限制更少的单目摄像头搭建体感交互系统逐渐成为可能。Juan C.Núñez 等[2]就提出了一种基于卷积神经网络（CNN）和 LSTM 循环网络组合，采用两阶段培训策略的手部关键点识别方法。虽然处理关键点识别的模型发展很快，也越来越准确，但也可以看到多数模型需要大量的计算和内存资源，无法快速部署在树莓派等边缘计算设备上，无法满足体感交互应用的便携性需求，限制了在小型设备上的应用。

对嵌入式设备的需求刺激了高效网络结构的发展，近年来出现了一系列轻量化卷积网络模型。GoogleNet[3]通过引入初始空间模块，以更低的计算成本更好地提取特征。MobileNets[4]中，Andrew 等人使用深度可分离卷积策略，将标准卷积分解为深度卷积和点态卷积，有效地减少了计算负荷。其中，ShuffleNet v2[5]使用分组卷积模式，引入通道分割和通道混洗操作，在较小模型规模的同时仍具有较强的泛化性，可以用比传统网络更少的参数保持相似的精度。因此，本文根据改进的 ShuffleNet v2 手部关键点回归模型，搭建了一个用于控制 MIDI 音乐设备的体感交互系统。系统以树莓派 4b 为搭载平台，加入二自由度舵机、无线通信等模块，实现了交互识别、动态跟踪、远程控制等功能，让 MIDI 设备控制更加直观自然、方便高效，使得体感交互系统易便携、低功耗、反应迅速。

1 基于改进 ShuffleNet v2 实现轻量化手势识别算法

手部关键点检测，也称为手部姿态识别，旨在定位手部关键区域，包括指尖、指关节等部位。为了在低成本的小型设备上完成体感交互的 MIDI 设备控制，以手部关键点检测为目标，从两个研究方向对模型进行改进：第一个方向是优化模型结构，强调图像中最重要的特征信息；另一个方向倾向于压缩模型，旨在以合理的精度损失减小模型。因此，本文以 ShuffleNet v2 为基础，增加 SENet 通道注意力机制从而提高精度，引入网络瘦身来降低计算消耗，搭建了一种轻量高效的手部姿态识别模型。

1.1 ShuffleNet v2 轻量级网络

ShuffleNet v2 是旷世在 2018 年提出的一种轻量级卷积神经网络，其设计遵循四个基本原则[5]：1）为了最小化内存访问成本，输入通道的数量和输出通道的数量应该尽可能相等；2）为了降低内存访问

[收稿日期] 2022－01－19

[第一作者] 周 祺（1990－），湖北武汉人，湖北工业大学教授，研究方向为产品创新设计

[通信作者] 张 帅（1996－），河北唐山人，湖北工业大学硕士研究生，研究方向为体感交互设计

成本，组卷积要尽可能小；3）为了提高网络并行度，网络结构要尽可能简单；4）为了减少运算消耗，激活等运算的次数要尽可能少。

为遵循其设计原则，ShuffleNet v2 网络采取了分组卷积的方式，在保证精度的情况下减少网络参数量。网络主要由两个类型单元构成：a 类型单元（图 1a）和 b 类型单元（图 1b），分别对应运算中步长为 2 和 1 的两种情况。

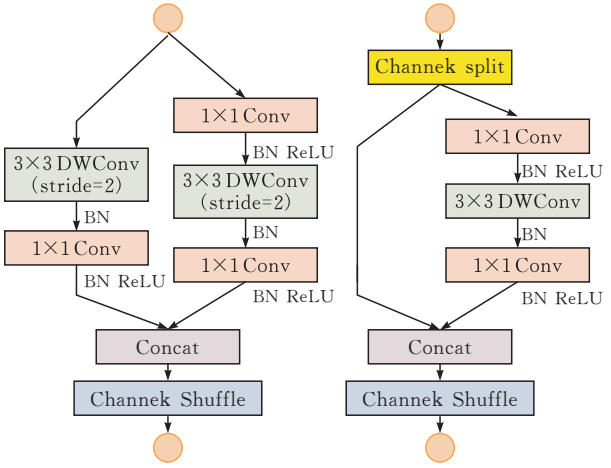
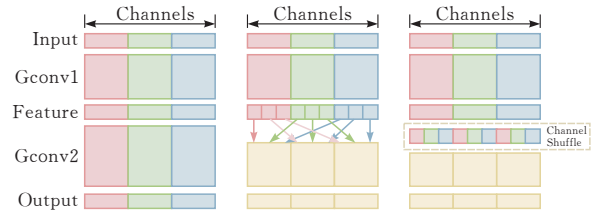


图 1 ShuffleNet V2 网络单元结构

a 类型单元为下采样单元，两个分支分别进行步长为 2 的 3×3 深度卷积（Depthwise convolution，DWConv）和 1×1 单元卷积操作。在卷积之后，两个分支通过级联操作（Concat）进行通道拼接合并，特征图维度大小减半，输出通道数加倍，最后进行通道混洗操作（Channel Shuffle）进行特征融合。

b 类型单元首先将输入特征通道 c 平均拆分为两个分支 c-c′和 c′，即进行通道分割操作（Channel Split）。为减少碎片化程度，在左分支 c-c′保持结构不变直接同等映射，右边分支 c′则按顺序进行 1×1 单元卷积、3×3 深度卷积和 1×1 单元卷积操作。通过级联操作拼接合并后，单元通道数保持不变，最后也需进行通道混洗操作。经过通道分割操作后，每次卷积计算都是在部分特征通道上进行的，计算量和参数相应减少，网络单元可以容纳更多的特征通道，提高了网络的准确率。

两个单元后的通道混洗就是在不同的组之后交换一些通道，从而交换信息，解决了分组卷积导致的信息丢失问题，使得各个组的信息更丰富，有利于提取到更多更好的特征（图 2）。通道混洗操作过程为：将输入层分为 g 组，总通道数为 g × n，首先将通道维度重塑为（g，n），然后将这输出特征转置变成（n，g），最后重塑为 g × n 进行输出。



(a)常规分组卷积 (b)特征图重组 (c)通道混洗操作

图 2 通道混洗操作

1.2 SENet 通道注意机制

SENet（Squeeze-and-Excitation Networks）通道注意机制是由 Hu 等[6]在 2018 年提出的，其核心思想在于通过学习特征通道的权重，使得有效的特征权重增大，无效或效果小的特征权重减小，能够达到更好的结果同时仅增加了可接受的少量计算代价。SENet 单元结构（图 3）主要有两部分，分别称为挤压（Squeeze）和激励（Excitation）。

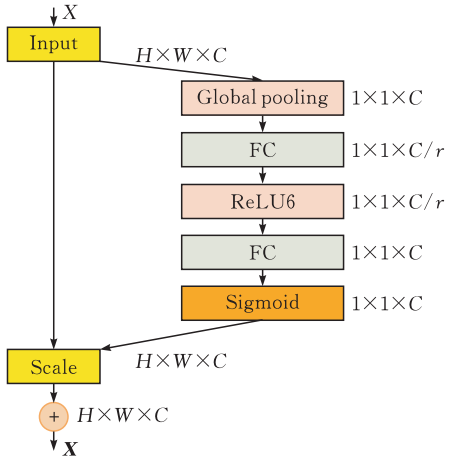


图 3 SENet 单元结构

对于卷积操作 $Ftr: X \rightarrow U$，$X \in \mathbb{R}^{H' \times W' \times C'}$，$U \in \mathbb{R}^{H \times W \times C}$，设 $V = [v_1, v_2, \cdots, v_C]$ 表示卷积核集，其中 v_C 表示第 C 个卷积核的参数；输出特征图有 C 个通道，即

$$U = [u_1, u_2, \cdots, u_C]$$

输入特征图有 C' 个通道，即

$$X = [x_1, x_2, \cdots, x'_C]$$

其中 u_C 可表示为：

$$u_C = v_C * X = \sum_{s=1}^{C'} v s_C * x_s$$

其中“*”意为卷积操作。

Ftr 可通过 SENet 单元如下操作来校准特征：原始特征图 X 首先进行挤压操作，通过全局池化（Global pooling）压缩到 1×1×C，将每个二维的特征通道变成一个特征标识符，在这种情况下 1×1 部分仍具有原始 H×W 感应野，并且将跨空间维度 H×W 的特征映射聚合了起来。全局池化操作生成的通道向量 $z \in \mathbb{R}^C$ 是由 X 的空间维度 $H \times W$ 通过收缩生成，其中 z 的第 c 个元素计算方法见

式(1)。

$$z_C = F_{sq}(u_C) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} u_C(i,j) \quad (1)$$

接下来进行激励操作，充分捕获通道依赖性，学习每个通道样本的特定激活，控制通道激活。激励层将挤压结果交由两个全连接层(FC)预测，对特征映射进行重新加权，通过 Sigmoid 函数的门机制把权值归一化，公式如下：

$$s = F_{ex}(z,W) = \sigma(g(z,W)) = \sigma(W_2\delta(W_1 z)) \quad (2)$$

式中，δ 为 ReLU 函数，$W_1 \in \mathbb{R}\ \frac{C}{r} \times C$ 和 $W_2 \in \mathbb{R}C \times \frac{C}{r}$ 表示两个全连接层生成的权重值，通过在非线性周围形成一个具有两个全连接层的瓶颈来参数化门控机制：W_1 中进行维度缩减，将 z 的维度从 C 缩减为 C/r，缩减后进入 ReLU 激活，之后在 W_2 中将维度再增扩回 C。

单元最终输入通过对参数的重新缩放转换获得，表示如下：

$$\tilde{x}_C = Fscale(u_C, s_C) = s_C \cdot u_C$$

其中 $\tilde{x}_C = [\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_C]$ 和 $Fscale(u_C, s_C)$ 指的是特征图中的 u_C 与激励操作产生的 s_C 之间的通道相乘。

1.3　轻量级关键点识别网络

手部关键点检测就是定位手部的关键点坐标序列，其中手部定位点序列(图 4)包括指尖，各节指骨连接处等 21 处关键点。以手部 21 处关键点序列为依据，由 Large-scale Multiview 3D Hand Pose 数据集和网络抓取的共 11200 张图片制作了训练图集。

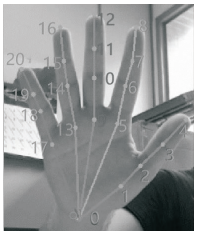


图 4　手部 21 点序列

训练中，为更好地模拟实际应用中的情况，在[−30，30]度之间随机旋转训练图像，将随机平移应用于边界框 15%内进行了中心扰动并进行了随机翻转。将大小为 256×256 的图像及关键点坐标输入调参后的 ShuffleNet v2 网络模型后，直接返回手部 21 个关键点坐标，调参后的网络结构见表 1。

为了在保证模型准确性的同时降低模型复杂度，提高泛化能力，在 Shufflenet v2 网络 b 类型单元右侧的最后一个单元卷积层之后添加 SENet 层，更改后的模块结构见图 5。ShuffleNet v2 网络结构加入的 SENet 单元，对提取的深度特征图进行重新校准，能够添加更丰富和更高级别的信息源，从而更好地引导模型的深度学习过程。

表 1　Shufflenet v2 关键点回归模型结构

阶段	输入	层	输出	副本数	步幅
1	256×256×3	Conv1 3×3	24	1	1
2	128×128×24	Shuffleunit	116	1	2
	64×64×116	Shuffleunit	116	3	1
3	64×64×116	Shuffleunit	232	1	2
	32×32×232	Shuffleunit	232	7	1
4	32×32×232	Shuffleunit	464	1	2
	16×16×464	Shuffleunit	464	3	1
5	16×16×464	Conv5 1×1	1024	1	1
6	16×16×1024	Avgpool	1024	1	—
7	1×1×1024	FC	21	1	—

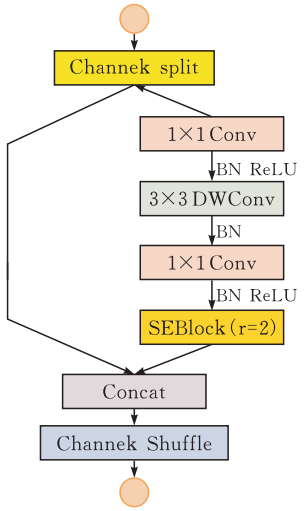


图 5　通道注意力机制 ShuffleNet v2 网络单元结构

1.4　模型剪枝与重构

虽然使用 ShuffleNet v2 构建的模型比较精巧，但仍然考虑进一步压缩模型以实现更快的推理速度，因此有必要对模型进行剪枝操作(图 6)。剪枝是一种常见的模型压缩方法，使用剪枝去除不重要的通道，可以减少神经网络的计算和内存需求[7]。

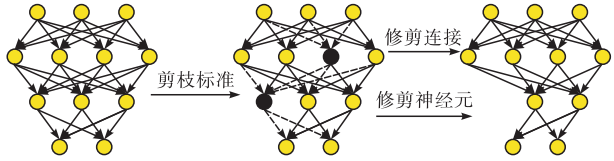


图 6　剪枝操作示意图

本文选用的网络瘦身操作是一种通道级剪枝方案，其基本思想是联合训练权重和引入的比例因子 γ 衡量每个通道的重要性[8]。网络瘦身根据 BN(Batch Normalization)层中的比例因子 γ 来衡量通道的重要性，可以指导模型在训练过程中将不重要的通道剪除，即 γ 较小时对应的通道。具体的网络瘦身流程为：首先在原始模型的 BN 层中加入比例因子 γ，将模型训练后通过系数 γ 的 L1 正则化约

束项来诱导 BN 层稀疏。然后通过比例因子 γ 的权重衡量通道的权重，找到可以丢弃的通道。最终训练修剪后的模型并将准确性恢复，构建更小的模型来移植参数。

网络瘦身方法的目标函数定义为：

$$L\text{slimming} = \sum_{(i,t)} l(f(i,W),t) + \lambda \sum_{\gamma \in \Gamma} g(\gamma) \quad (3)$$

公式(3)为调整后的模型损失函数，(i,t) 为训练输入和目标，W 为网络中的可训练参数，即 $\sum_{(i,t)} l(f(i,W),t)$ 为原始模型的训练损失函数，后半部分为用于约束的比例因子 γ，g(.) 是比例因子上的惩罚项，λ 是两者的平衡因子，g(.) 使用 L1 正则化，即 $g(s)=|s|$。L1 的正则化使得 BN 层的比例因子趋近于零，能够识别不重要的通道，有助于后续的通道剪枝，甚至可能提了泛化精度。

训练时，根据 ShuffleNet v2 的结构特点，仅针对步长为 1 的 ShuffleNet v2 单元右分支进行网络瘦身操作，修剪了 50%的通道。重构的小网络经过微调步骤移植模型参数，最终模型大小从原始模型的 5.4 MB 降至 3.9 MB，用于衡量模型复杂度的浮点运算次数(Floating point of operations，FLOPs)从 2.36×10^7 降至 1.77×10^7，参数量减少 25%。当手部关键点训练集在经过修剪后的关键点检测模型上将实现较好的识别准确度，即经损失函数计算所得准确率大于 90%后，将其部署至树莓派设备上。

2 体感交互系统设计

用于 MIDI 设备控制的体感交互系统由树莓派、摄像头、二自由度云台、Arduino、红外传感模块、nRF24L01 通信模块和 5 针 DIN 插座组成。系统分为两个部分，以树莓派 4b 作为数据处理端，Arduino uno 作为数据接收端。树莓派作为上位机，通过摄像头获取的图像计算分析关键点位置，控制二自由度云台追踪手部，并将手部姿态信息通过通信模块传递给下位机 Arduino uno，从而实现对 MIDI 设备的体感控制，模块框图见图 7。

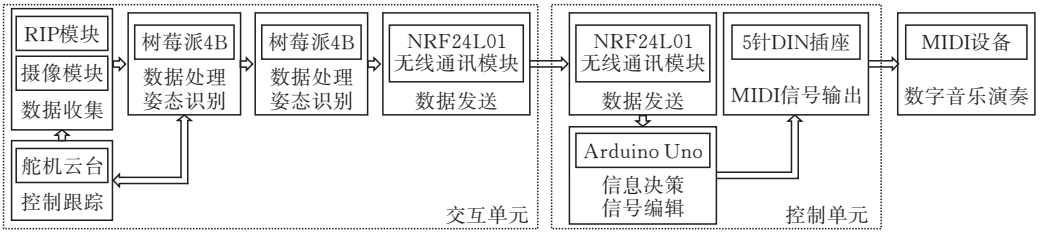


图 7 体感交互系统模块框图

2.1 舵机追踪模块

为了让用户在交互时手掌处于摄像头中心位置，使用二自由度云台对手部中点进行追踪。为实现舵机角度的平滑调整，采取了 PID 控制，即比例(proportional)－微分(integral)－积分(derivative)控制修正系统偏差。追踪控制以离散形式实现，需要采用后向欧拉方法对控制器方程进行数值积分[9]，计算公式为：

$$u(k) = K_P e(k) + K_I \sum_{i=0} e(i) + K_D[e(k) - e(k-1)]$$

其中 K_P、K_I 和 K_D 分别是比例系数、积分系数和微分系数。

根据手部识别模型返回手部 21 个关键点的坐标集，可得手部中心坐标为 (x_h, y_h)，图像中心点坐标为 (x_c, y_c)，则第 i 张图片图像中心点与手部中心点纵向和横向偏差分别为：

$$\Delta x_i = x_h - x_c \text{，} \Delta y_i = y_h - y_c$$

横向舵机方位角有效范围为 0～180°，纵向舵机方位角有效范围为 0～90°，因此追踪模块中最终所用的舵机移动角度计算公式为：

$$a_x = K_P \Delta x_i + K_D(\Delta x_i - \Delta x_i - 1) \text{，}$$

$$a_y = (K_P \Delta y_i + K_D(\Delta y_i - \Delta y_i - 1))/2$$

舵机的控制部分需要使用 PWM(脉冲宽度调制技术)，利用占空比来控制脉冲信号的输出大小，靠脉冲信号的持续时间来定位舵机输出轴的旋转角度。系统选用舵机的 PWM 频率为 50 Hz，转动范围为 0～180 °，对应的 PWM 周期 T 为 20 ms，其脉冲长度 t、占空比 D 和转动角度之间的对应关系如图 8 所示。

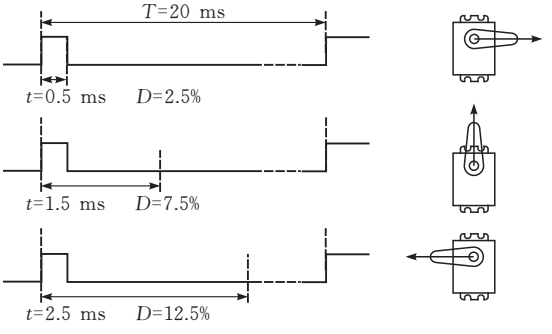


图 8 脉冲长度与转动角度对应关系

为防止舵机追踪抖动造成镜头不稳，模块中设置一个 24×24 的死区，并使用多线程进行横滚轴和俯仰轴的 PID 角度修正运算，从而达到及时稳定的手部跟随效果。系统检测到交互区域内有红外信号时初始化舵机和摄像头，舵机转至初始位置；在时限

内监测区域未识别到目标手势信息时,舵机回归初始位置并在待机时间结束后释放相关端口以节约算力,控制流程见图 9。

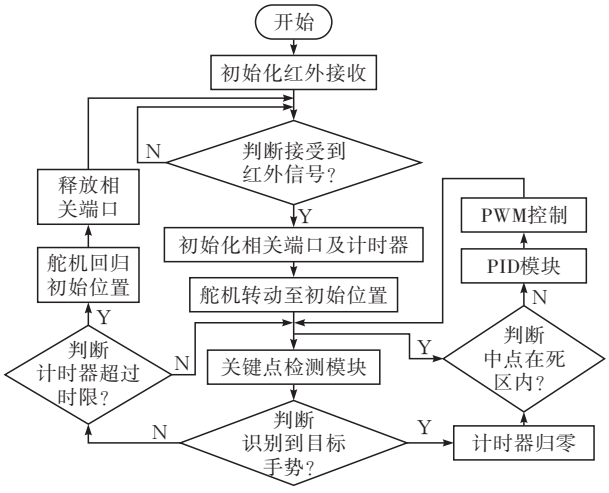


图 9　舵机控制流程

2.2　无线通信与控制模块

上位机通过 nRF24L01 单片射频收发芯片将识别到的动作信息传递给同样配备 nRF24L01 通信模块的下位机,实现中远距离的无线通信。对 nRF24L01 通信模块的地址、通信频道等在收发端进行配置,实现多机的数据传输,可以组建星状控制网络,方便用户同时控制多个 MIDI 设备。

控制端主要完成以下流程(图 10):1)接受手部姿态信号;2)识别姿态编号并转换为 MIDI 消息;3)通过 5 针 DIN 插座输出信号控制 MIDI 设备。

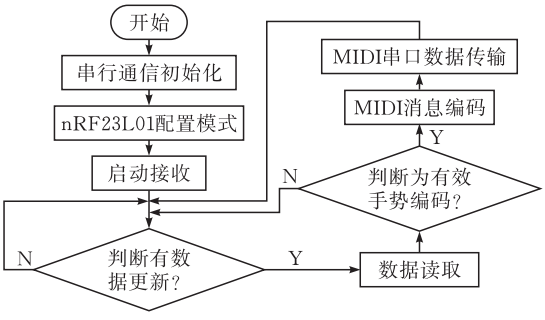


图 10　控制端工作流程

5 针 DIN 接线传递的 MIDI 控制消息由十六进制符号表示,由控制器号和数据字节共同组成。控制器号大于 128,间于 0x80 到 0xFF(十六进制);数据字节小于 127,间于 0x00 到 0x7F(十六进制),控制端组合发送给 MIDI 设备后可实现音量更改或音高变化等功能[10]。

3　实验测试与结果分析

进行系统测试时,数据处理端树莓派 4b 运行姿态识别模型及传达指令,摄像头 Camera V2 实时采集用户图像信息。Thonny Python 作为开发环境运行手势识别模型文件,在识别出手部关键坐标后,通过计算其二维角度关系识别手势。Arduino uno 作为控制端,通过 USB-MIDI 接线与 PC 机相连,使用 MIDI-OX 程序监控传入的 MIDI 数据,测试对应 MIDI 音色的演奏情况(图 11)。

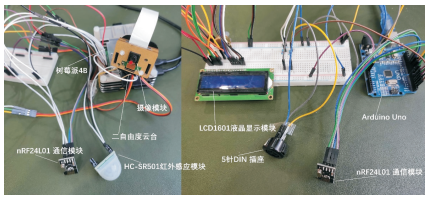


图 11　测试系统构成

在测试时,共定义了 10 种手部控制姿势,包含 4 个简单方向手势(SG),4 个简单手指手势(FG)和 2 个精细复杂手势(CG)(图 12),分别对应不同的 MIDI 信号。每个动作分别在 ShuffleNet V2 网络在网络瘦身前后的模型上进行测试,获取得到识别准确率、帧率及响应时间(表 2)。

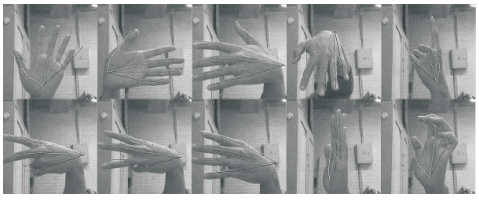


图 12　测试手势示意图

表 2　shufflenet v2 关键点回归模型结构

	准确率/%											帧率/(f·s^{-1})	响应时间/ms
	SG1	SG2	SG3	SG4	FG1	FG2	FG3	FG4	CG1	CG2	总计		
原模型	92.5	93.3	92.5	90.8	80	79.2	82.5	82.5	63.3	57.5	81.4	8.78	224
网络瘦身模型	91.7	92.5	92.5	90.8	80	78.3	79.16	80	58.3	52.5	79.5	10.12	192

手势交互实测中,识别准确率在原模型上的平均准确率为 81.4%,网络瘦身后模型的平均准确率 79.5%。与原模型相比,其准确率只下降了 1.9%,在简单方向手势上其相差仅为 0.8%,手势识别的准确率在前后并未大幅下降。在响应时间上,原模型和剪枝模型都实现 8 帧/s 以上的运算帧率,并且剪枝后的模型速度提高了 14.3%,对比原模型能更及时地完成信息反馈,有利于用户实时控制 MIDI 设备的演奏。总体来说,修建后的模型,在基本不影响准确性的情况下,其响应速度和模型大小都优于原模型,能够及时完成 MIDI 音乐的体感交互操作。

4 结论

改进的 ShuffleNet v2 模型实现了对更小、更快的追求，基本能够同步完成 MIDI 设备的体感交互任务。针对具有精度要求和时间敏感的体感交互应用，本系统为在小型化设备上进行关键点推理提供了一个解决方案。结合无线通讯技术和 MIDI 音乐标准，实现了远程无接触手势控制数字音乐演奏的相关功能。系统如果更新关键点二维角度算法，还可增加及修改交互动作，扩展其手势识别库。研究结果为体感交互和关键点识别的实时连续识别和纵向扩展开辟了一个有趣的实践方向，为 MIDI 音乐创作者提供了多样化的交互方式，且便于携带、成本不高、易于拓展。

[参 考 文 献]

[1] LUTHER L, TIBERIUS V, BREM A. User Experience (UX) in business, management, and psychology: A bibliometric mapping of the current state of research [J]. Multimodal Technologies and Interaction, 2020, 4(02): 18.

[2] NUNEZ J C, CABIDO R, PANTRIGO J J, et al. Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition [J]. Pattern Recognition, 2018, 76: 80-94.

[3] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]. // Proceedings of the IEEE conference on computer vision and pattern recognition, 2015:1-9.

[4] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[EB/OL]. (2017-04-17)[2021-12-30]. https://arxiv.53yu.com/abs/1704.04861.

[5] MA N, ZHANG X, ZHENG H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]. // Proceedings of the European conference on computer vision (ECCV), 2018: 116-131.

[6] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C]. // Proceedings of the IEEE conference on computer vision and pattern recognition, 2018: 7132-7141.

[7] 林景栋，吴欣怡，柴毅，等.卷积神经网络结构优化综述[J].自动化学报，2020，46(01)：24-37.

[8] LIU Z, LI J, SHEN Z, et al. Learning efficient convolutional networks through network slimming [C]. // Proceedings of the IEEE international conference on computer vision, 2017: 2736-2744.

[9] BALAJI V, BALAJI M, CHANDRASEKARAN M, et al. Optimization of PID control for high speed line tracking robots[J]. Procedia Computer Science, 2015, 76: 147-154.

[10] DE OLIVEIRA H M, DE OLIVEIRA R C. Understanding MIDI: A Painless Tutorial on Midi Format [EB/OL]. (2017-05-15)[2021-12-30]. https: // arxiv.53yu.com/abs/1705.05322.

Design of MIDI Somatosensory Interaction System based on Hand Posture Recognition

ZHOU Qi, ZHANG Shuai

(*School of Industrial Design, Hubei Univ. of Tech., Wuhan* 430068, *China*)

Abstract: In order to get rid of the limitation of MIDI music equipment in the way of interaction, and to solve the problems of poor portability and high price of the current somatosensory interaction system, combining edge computing equipment with a lightweight network, a music somatosensory interaction system based on hand gesture recognition is proposed. The host computer of the system relies on the Raspberry Pi 4b, obtains data through the monocular camera to identify the key points of the hand, calls the steering gear to track the target, and sends the action command to the lower computer through the nRF24L01 communication module, so as to realize the medium and long-distance control of music performance. By embedding the SENet channel attention mechanism in the lightweight network ShuffleNet v2 and performing the network slimming operation, the system can complete the somatosensory control task within 200ms, which can meet the needs of user gesture control, device coordination, etc., and the system is convenient and efficient. It provides a reference for deploying somatosensory interactive applications on mobile devices.

Keywords: MIDI; Raspberry Pi; gesture recognition; ShuffleNet v2; somatosensory interaction

[责任编校：张岩芳]