基于图像识别和神经网络技术的影视声音后期工艺探索

郭境熙 刘 嘉

（北京电影学院声音学院，北京 100088）

【摘要】本研究选取实际工作中最常见的脚步编辑工作作为对象，基于开源的 Openpose 框架分析提取人物在画面中的运动姿态，进而利用支持向量机（SVM，Support Vector Machine）和多层感知器（MLP，Multilayer Perception）这两种神经网络学习方法分别对 Openpose 的数据进行处理，训练出针对于不同画面和运动情况的预测模型。在此过程中，对比了不同的预处理组合对模型拟合的影响，最终将其结果编码为多种音频工作站所能兼容的标准 XML 序列，尝试和探索在习惯、精度、实用性等方面均可接受的一种声音后期工艺的辅助手段。

【关键词】声音编辑 图像识别 神经网络 机器学习 姿态推测

【中图分类号】J933

1 引言

随着影视文化产业的繁荣，社会对影视产品的需求也越来越大。今天，由于工业化技术水平的提高，影视声音后期制作的分工越来越细，质量也越来越好。无论是院线电影、电视剧，还是网络平台上播放的一些剧集，甚至包括目前非常流行的抖音小视频，都很明显地感受到了这种趋势。然而，随着节目数量爆发性地增长，随之而来的问题是有限的制作劳动力资源与越来越多的节目数量和质量需求之间不可调和的矛盾。从目前的工艺来看，声音编辑环节在整个制作流程中占据了绝对的时间比例，它有着人力高度密集、工作强度大、时间占比长、重复性劳动非常多的特点，在实际工作中，从业人员大量的时间和精力会耗费在对声音素材的检索、分类、编辑、声画同步等工作上。与此同时，我们看到声音编辑工作并非简单劳动，一个合格的声音编辑人员往往需要经过长时间的职业技能培训，才能满足这种艺术产品的生产质量要求。所以，无论是采用简单地增加个体劳动时间，还是快速扩大声音编辑从业人员的规模来满足越来越多、越来越高的行业制作需求，都是不现实的。

从 2015 年起，人工智能的研究和应用逐步开始进入到大众的视野，在图像识别领域和神经网络系统领域不断出现了一些有趣的思路和新的应用场景。得益于算法的发展以及计算能力的提升，图像识别的准确度相对以前有了很大的提高，近年来 ImageNet 的年度大规模视觉识别挑战赛中（ILSVRC[①]），排名前五的深度学习神经网络错误率即能做

【作者信息】郭境熙（1996－），男，学士，纽约大学 Tisch（蒂势）艺术学院在读硕士，主要研究方向：交互式通信；刘嘉（1970－），男，博士，北京电影学院声音学院副教授，主要研究方向：影视声音创作/影视声音技术。

到小于5%。这其实已经在分类准确率上高于人类了，而随着于计算机硬件的快速迭代和性能提升，在目前普通家用计算机平台上利用性能稍好一点的GPU和CPU就能轻易地满足以上视觉识别和神经网络深度学习的需求。

在我们通常的习惯思维中，影视声音后期往往被更多地认为是一个艺术创作过程，技术性的手段被限定在了一个很有限的范围里。并且由于声音自身的特性以及基础研究长期停滞不前，造成了它几乎是现在影视制作中自动化和智能化程度最低的一个领域。但是，如果我们理性地对整个过程进行结构化的回溯，就会发现其实前面提到的很多工作内容，尤其在重复性地耗费大量精力进行声画同步的声音编辑过程中，可以将其分解为画面同步点的判断、声音素材时间线位置确定和声音素材的放置三个环节。假如采用现有的图像识别和神经网络学习技术建立标准模型，实现素材的自动调用，则有可能实现整个过程或者过程中某些重要环节的智能化辅助。

鉴于此，笔者以影视作品中最常见的脚步声编辑作为研究对象，在过程中利用开源的Openpose（人体关键点实时检测）框架对画面人物的动作进行识别，将结果数据进行必要的预处理后同时用SVM（支持向量机）和MLP（多层感知器）进行训练，观察得到的训练模型的结果，并生成标准的时间线XML，导入工作站进行素材链接，从而实现基本的声音编辑过程中声画同步的自动化。

2 研究工具与对象的选取

脚步声的编辑过程，我们按照工作流程把它分为了四个步骤：（1）在画面中找出人物的脚部并观察鞋、地的材质用以确定声音素材的选择；（2）在时间线上找到左右脚触地的精确位置（时间点）；（3）从声音资料库中检索符合要求的素材；（4）将声音摆放到时间线上并做相应的微调。可以发现在整个过程中重复性劳动最多，耗费时间最长的是在步骤2和步骤4上，同时，由于步骤1和步骤3目前还受制于相关基础元数据的智能化标记完善程度制约，所以，笔者暂时对这两个步骤进行简化（将在后文说明），事先根据所训练的对象准备好一套适合的脚步素材库以备调用，而把主要精力放到通过机器学习和神经网络算法将画面中脚步在时间线上的精确定位（步骤2）和声音素材的自动套用上。

实验过程中首先借助开源框架Openpose来进行画面人物对象的识别和人物姿态推测。Openpose是一个由卡内基梅隆大学维护的人体姿态识别开源库，基于CAFFE框架开发，以卷积神经网络和监督学习为手段在不通过任何传感器的帮助下，实现画面中一个或多个人物的姿态识别。目前，可以完成：单人最多25个身体/脚部关节的关键点识别（本研究选取方案）；或42个关键点的手部关节识别（左右手各21个）；或70个面部关键点的识别，特别是身体关键点识别模式下，运算时间与检测出的人数无关，这无疑为大规模的部署和真实应用环境中复杂的画面提供了非常强大的分析工具。

在得到关键点数据集之后，通过人工的方法对左右脚落地的时间进行标记和其他的预处理，构成下一步模型训练的基础。我们期望将原始数据和标记后的数据通过机器学习对二者进行拟合，最终训练得到一个可以通过每帧画面人体关节点坐标关系判定脚步落下的模型，用于实际工作中画面的识别。在这个环节利用了目前比较常见的两种神经网络算法：支持向量机（Support Vector Machines，SVM）和多层感知器（Multilayer Perceptron，MLP），我们在过程中观察各自的表现，比较它们的结果并评价其差异。

经过训练和拟合后的模型可以识别画面中的人物，检测画面中人物的脚步落下时刻，最终将结果通过脚本导出XML文件，利用非线性编辑工作站实现脚步声音素材在时间线上的自动放置。

3 数据的准备与数据处理流程

训练样本的获取使用了GH5相机以1080p/60fps的格式拍摄下几种不同状态的单个人物运动视频片段，包括了两种机位方式以及四种维度的移动（表1），以保证人物运动的多样性。之所以拍摄单人而非多人，是考虑到在实验初期尽可能控制一些不必要的变量，实际上，Openpose对同一画面中

多人的姿态判定准确程度与单人基本没有差异，如果最终训练结果模型能适用，则多人画面的处理仅仅为计算量的简单放大而已。此外，最初考虑60fps的原因是与时分秒的进制数一致，为某些环节中可能出现的人工修正减少一些误差。但是，这些视频测试数据量的提高在实验中对于模型准确性反而造成了不必要的扰动。经过观察发现更低的帧率可以一定程度上解决上述两个问题，不但提高精度，而且还降低计算成本，因此选择将所拍摄画面的帧率和分辨率等倍缩小成720p/30fps的数据。

表1 拍摄内容

固定机位从左向右	固定机位从右向左	固定机位背对相机向前	固定机位面对相机向前	固定机位四个方向原地踏步
跟踪人物从左向右	跟踪人物从右向左	跟踪人物背对相机向前	跟踪人物面对相机向前	

3.1 流程（图1）

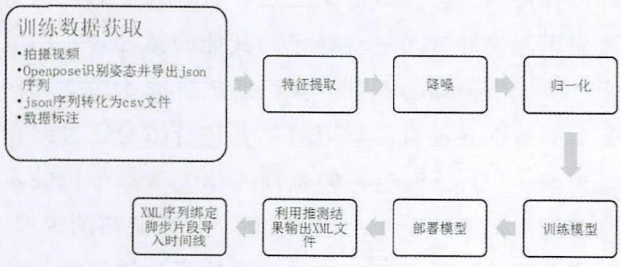

图1 数据处理流程图

3.2 Openpose 框架

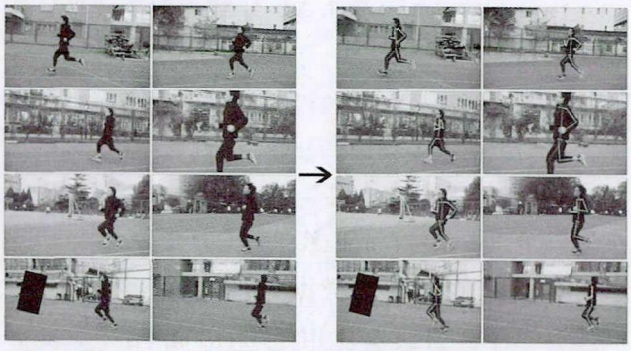

图2 左边为原始视频的缩略图，
右边为经过Openpose推断后视频的缩略图

使用Openpose对拍摄画面进行分析和姿势判定，设定最大推断人数为1。在过程中人物的骨架信息被嵌入到输出画面中（图2），并以单帧的形式输出人物骨架的关节点信息。推断结束后，得到带有骨架图像的视频和连串的json序列，每个json文件中中包含人数、身体部位等属性以及关节数据的一维矩阵。这个一维矩阵包含了25个关节点的x，y值绝对坐标值，以及推断对应关节点x，y值的权重系数，共75个特征参数。

3.3 json序列转化为csv文件

Openpose输出的数据是基于帧的单个json文件，在训练模型时，需要多次读取数据，这会耗费不必要的时间，同时，当把时序信息作为训练特征的一部分时，单帧输入的数据会造成不利于模型训练的问题。因此在此次试验过程中，各个视频导出的json序列里的一维都被逐个提取出来，各自存放在一个csv文件中，最终每个视频得到的数据序列是一个二维数组，行是单帧的关节点信息，列为帧序列信息。

图3 json文件序列转换为单个csv文件

3.4 数据标注

在得到数据之后，首先需要对原始数据进行以帧为精度的标注，神经网络才会根据标注的值去“学习”当前特征数据所处的状态，并通过前向传递与反向传递算法拟合这批特征数据形成模型。另外需要特别提及一点，在此进行数据标注所选用作为标注依据的关节点，与

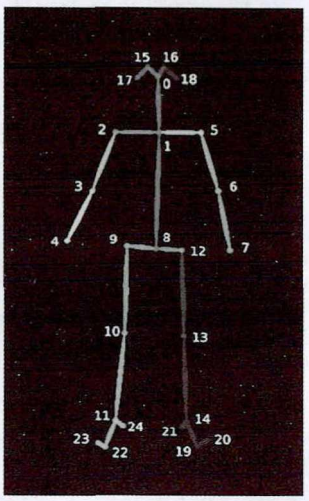

图4 关节坐标序号对应图

模型训练过程中所选择的关节点没有直接的关系，模型拟合的趋势是由特征值自己本身数据导向的。

实现思路如下：将人物的运动抽象为左脚周期与右脚周期，左脚周期开始于左脚落到地面上的对应帧 f_i，左脚周期结束于右脚落到地面上的对应帧 f_{i+n}，并开始右脚周期，例如刚刚例子中左脚周期的帧数为 n。图 4 是骨架关节点的号码对应图，本次研究中笔者使用 $[j_0 \sim j_{24}]$ 来表达 0～24 号（共 25 个）关节点。左脚参照的关节点选择 $[j_{13}, j_{14}, j_{21}, j_{19}, j_{20}]$ 作为标注依据，右脚参照的关节点选择 $[j_{10}, j_{11}, j_{24}, j_{22}, j_{23}]$ 为标注依据，当左脚 $[j_{14}, j_{21}, j_{19}, j_{20}]$ 关节点接触地面时，开始右脚周期，$[j_{11}, j_{24}, j_{22}, j_{23}]$ 关节点接触地面时，结束右脚周期，开始新的左脚周期。左脚周期中所包含的所有帧都用 0 作为标注值，右脚周期中所包含的所有帧都用 1 作为标注值，最终图像脚步识别的问题就被简化为二分类问题。

通过纯图像识别关节点的位置会有轻微波动，在标注过程中，所用到的关节点都作等权重处理，所有关节点都会被用于状态判别，这样能缩小姿态推断过程中骨架数据的轻微波动带来的影响。

9.48E+02	6.34E-01	6.01E+02	9.25E+02	9.05E-01	1.00E+00
9.45E+02	6.35E-01	5.98E+02	9.19E+02	8.75E-01	1.00E+00
9.48E+02	6.74E-01	5.95E+02	9.22E+02	7.91E-01	1.00E+00
9.45E+02	6.80E-01	5.86E+02	9.16E+02	7.70E-01	1.00E+00
9.45E+02	6.58E-01	5.77E+02	9.19E+02	8.10E-01	1.00E+00
9.48E+02	6.99E-01	5.74E+02	9.22E+02	7.98E-01	1.00E+00
9.48E+02	6.93E-01	5.69E+02	9.19E+02	7.37E-01	1.00E+00
9.51E+02	7.76E-01	5.54E+02	9.28E+02	7.65E-01	1.00E+00
9.48E+02	7.51E-01	5.51E+02	9.25E+02	8.42E-01	1.00E+00
9.51E+02	7.65E-01	5.45E+02	9.28E+02	7.87E-01	1.00E+00
9.51E+02	7.97E-01	5.30E+02	9.28E+02	7.96E-01	1.00E+00
9.51E+02	7.91E-01	5.27E+02	9.28E+02	8.30E-01	1.00E+00
9.54E+02	7.85E-01	5.10E+02	9.28E+02	7.79E-01	1.00E+00
9.51E+02	8.02E-01	5.04E+02	9.25E+02	7.74E-01	1.00E+00
9.54E+02	7.54E-01	4.95E+02	9.25E+02	7.64E-01	1.00E+00
9.54E+02	7.74E-01	4.83E+02	9.19E+02	8.22E-01	1.00E+00

图 5 方框内为手动标注内容，“1”代表这部分数据正处于右周期

3.5 数据预处理

经过 Openpose 获得的原始数据能提供 75 个输入特征，关节点的坐标数值较大，各关节点 x 值的区间在 [0，1280]，y 值区间在 [0，720]，推断过程中的权重值 c 的区间在 [0，1] 之间，各点关系是离散的，这对模型拟合有极大影响（若不经任何处理，直接将原始数据用于训练，模型在测试集上表现正确率为 40%～60%）。为了提高模型的精确度，在数据输入之前，需要提前做一些预备处理，此外，考虑到使用的机器学习模型包括 SVM 和 MLP，这两者对输入特征的数据相对敏感，亦需要人为减少无用特征。

3.5.1 特征提取

在训练开始前，需要预估某些特征值的作用，并将无用的特征值删除。在本次研究中，对于人物脚步，由 Openpose 推断所得的权重值 c 并无用处，经过实验和观察（可参考本文 4.2），代表 x 值的特征以及头部的特征对模型拟合没有太大的意义，因此也将其删除。

在本次研究中，笔者并没有加入深度学习的算法，MLP 或者 SVM 算法很难对画面内部信息进行特征提取。每一个关节点都是绝对的坐标值，意味着人物在画面中从左向右移动，关节点的 x 值会逐渐增大，人物在画面中上下移动时，y 值会产生一些噪声。通过下面的方法，可以将 x，y 的绝对坐标值转化为相对坐标值：

$$x_{ref} = \frac{(x_1 + x_8)}{2}$$

$$x_{i'} = x_i - x_{ref}$$

$$y_{ref} = \frac{(y_1 + y_8)}{2}$$

$$y_{i'} = y_i - y_{ref}$$

x_1 表示关节点 1 坐标的 x 值，x_8 表示关节点 8 坐标的 x 值，y_1 表示关节点 1 坐标的 y 值，y_8 表示关节点 8 坐标的 y 值（图 4），x_{ref}，y_{ref} 表示计算的出的参照点对应的 x，y 值，y_i 表示原始绝对坐标值，$y_{i'}$ 表示的是相对坐标值。关节点 1 与关节点 8 的中间连线代表了人物骨架的躯干部分，处于整个骨架

的中心位置，所以将躯干部分的中点设置为参照点。这种处理方式可以近似看作将人物从画面中分割出来，并得到各关节点于参照点的相对位置（图 6）。

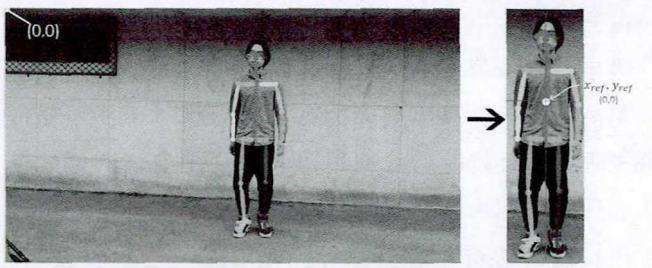

图 6 将人物从画面中分割，获得参照点

3.5.2 算数插值

由于 Openpose 推断出人物画内部分的姿态，人物身体部分被遮挡或者超出画面时，这部分关节点的数值会被 0 填充，导致这部分的数值变成离散值，亦会影响模型的拟合。故使用以下方法对缺失帧 f_n 进行填充：

$$f_n=\frac{(f_{n-2}+f_{n+2})}{2}$$

其中，f_n 表示第 n 帧的数值，f_{n-2} 表示第 n－2 帧的数值，f_{n+2} 表示第 n＋2 帧的数值。需要特别说明的是，在研究阶段的初期，希望尽可能研究典型状态得到收敛结果，对于在长时间段的关键点缺失问题暂时排除在本次研究范围之外，因此采用了最简单的插值算法，只补充 2 帧以内的数据缺失。这对于目前的数据集是可行的，但是一旦出现 3 帧或以上的数据缺失，这种降噪方式范围反而会给整个数据集加入更多的干扰噪声。另外，插值算法即使补全了缺失的值，依旧会对模型的准确度产生影响。

3.5.3 归一化

归一化是一种线性变化，可以将样本的特征值映射到［－1，1］或［0，1］的区间内。这种变化对数据改变合并不会造成数据失真，还能提高数据在模型拟合中的效率，而且这种变换不会改变原始数据的数值排序。数值归一化以后，能加快梯度下降的求解速度，而且避免了因为特征值过大权重值偏移而导致过拟合。

骨架中的各个关节点对应的坐标所在的数值区间各不相同，在此研究中，经过得到各个关节点与 x_{ref}，y_{ref} 的相对坐标值，例如头部的坐标点和脚步的坐标点的相对距离会更大，肩部坐标点与胯部坐标点的相对距离会比前者要小，手部的坐标点与胸部坐标点的相对距离甚至会更小。不同特征的数值区间分离过大时，在模型训练的过程中每个参数学到的权重值很可能会偏移，所以需要通过归一化将各个特征值归一到一个相对数值比较小的区间中。

在本次研究中使用了均值归一化：

$$m'=\frac{m-\overline{m}}{m_{max}-m_{min}}$$

方法（1）Mean Normalization

m' 表示归一化后的结果，$\overline{m}$ 表示单个特征值的平均值，m_{max} 代表特征序列中的最大值，m_{min} 代表特征序列中的最小值。

在实验的过程中还尝试了另外两种归一化方法：

$$m'=\frac{m-m_{min}}{m_{max}-m_{min}}$$

方法（2）Min-Max Scaler

$$m'=\frac{m-Q_1(m)}{Q_3(m)-Q_1(m)}$$

方法（3）Robust Scaler

Robust Scaler 用到了四分位数的思路，Q_1（m）等于该样本中所有数值由小到大排列后排在 1/4 位置的数字，Q_3（m）等于该样本中所有数值由小到大排列后排在 3/4 位置的数字。

当前研究中输入特征为稠密矩阵，各个输入特征数值差距比较大，后两种归一化方法在原理上不太适合，且经过多次对比测试后，发现使用均值归一化处理过的数据做训练的模型，推断结果的正确率平均高出 2%～3%。

另一方面，在画面中人物也会在画面纵深方向移动，这意味着在同一段视频内，如果人物在其中是纵深移动，点与点的绝对距离会根据近大远小的规则变化，这相当于也在时序轴上加入了不可忽略的噪声。在输入数据之前，通过如下一种归一化方法将这部分噪声去除：

$$|j_1j_8|=\sqrt{(x_1-x_8)^2+(y_1-y_8)^2}$$

$$x_{i''},\ y_{i''}=\frac{(x_{i'},\ y_{i'})}{|j_1j_8|}$$

$|j_1j_8|$表示关节点1、8之间的距离，也就是躯干部分的长度。在画面内人物远近移动时，身体各部分可视作在同一焦平面上，即各个部分点与点之间的距离比值几乎是一致的，因此可将$|j_1j_8|$用作缩放系数。每一帧各个点与参考点的距离经过除法运算以后，由于远近而带来的数值波动能较好消除。

4 模型选择与训练

4.1 训练工具与模型的选择

本次试验中，笔者选择scikit-learn[②]这个机器学习库。scikit-learn是一个python的机器学习库(以下简称为sk-learn)，提供了多种机器学习模型以及数据处理方法。在训练中我们从sk-learn库中选取了两种机器学习任务中较为常用的模型：支持向量机（SVM）和多层感知器（MLP)。此外，笔者借助numpy库的numpy.array作为数据在训练过程中的承载方式，所有的操作都以numpy.array的形式操作。

训练过程中以一帧作为一个单位样本，目的在于通过每一帧的关节坐标信息来判定画中人物脚步所在的运动周期是左还是右。在训练过程中将人物的运动方式分为以下几类分别进行训练：

(1) 固定机位，四向人物位置固定的脚步运动；

(2) 固定机位，人物向纵深位置的脚步运动；

(3) 固定机位，人物在画面中从左向右/从右向左移动；

(4) 机器跟随人物，人物正面向前移动；

(5) 机器跟随人物，人物背面向前移动；

(6) 机器跟随人物，人物侧面向前移动。

4.1.1 支持向量机（SVM)

SVM是在分类与回归分析中分析数据的监督式模型与学习的二元分类的广义线性分类器，它的决策边界是对学习的样本求解最大边距超平面。学习权重值分别选择[0.1，1，10，100，200]。

4.1.2 多层感知器（MLP)

MLP是人工神经网络的一种，使用计算机构成一个一个的神经元，多个单元组成单层的神经层，再由多个神经层连结起来组成多层的神经网络。神经网络的设计思路来源于人的大脑皮层，以此模仿人脑的思考方式。

在全连接神经网络的实验中，共有20个输入单元，3个隐藏层，各个隐藏层的神经元个数分别为[20，20，10]，输出单元为1个，激活函数为ReLU，学习速率分别选择a=[0.001，0.01，0.1，0.5]。由于样本数量比较少，在训练过程中一次性将所有的样本都用于模型的训练，不设置批次大小。

4.2 训练流程

在训练开始前，所有的数据都会采用逻辑回归模型用于基本二分类收敛测试，首先观察通过逻辑回归处理的数据集模型是否有收敛倾向，确认有数据集有收敛倾向，再考虑进行剩余模型的训练。由于实验是二分类问题，正确率高于50%+10%即视为有收敛倾向。

在训练开始时，先将csv文件导入到python项目中，使用numpy.array构建一个矩阵承载数据，矩阵的结构与csv文件数据的结构是一致的，行为单帧的关节点信息，列为帧序列信息。

sk-learn提供的MLP与SVM（在SVC类下）方法获得训练数据的方式是一样的，因此不需要单独为这两种模型重新构建数据结构。模型会将每一行视为单帧样本，将每一行的各个元素视为一个输入特征，当第一个样本拟合结束后，会自动跳入到下一个单帧样本，读取下一个样本的元素。我们将已标注的数据集分成两类：一类叫做训练集，另一类叫做测试集。训练集用于MLP与SVM模型的训练，测试集用于评判模型拟合的效果，得出正确率的值可以作为预处理方法和模型鲁棒性优劣的参考。

4.3 几种不同的思路

在模型训练过程中，尝试了组合不同预处理方法的训练集。各种组合对于训练结果的正确率影响不同，其中包括对x，y输入特征和各个关节点选择的取舍，取舍原因在文中3.5提及。根据观察判断，与下半身相关的关节点会对结果产生直接影响，而脚步运动在画面中也可以抽象为垂直运动，亦可以预测y值会比x值重要。根据多次实验得到的结果

确实可观察得出，经过降噪与归一化后的数据用于处理有利于模型拟合。

在实验中所用的数据表示见表 2。

表 2 缩写及对应输入特征关节点

缩写	输入特征
[x，y]	x，y 值
[y]	y 值
[10 joints]	[j_{13}，j_{14}，j_{21}，j_{19}，j_{20}，j_{10}，j_{11}，j_{24}，j_{22}，j_{23}]
[8joints]	[j_{14}，j_{21}，j_{19}，j_{20}，j_{11}，j_{24}，j_{22}，j_{23}]
[left/right joints]	[j_{13}，j_{14}，j_{21}，j_{19}，j_{20}] 或 [j_{10}，j_{11}，j_{24}，j_{22}，j_{23}]
[20 joint]	除去 [j_{17}，j_{15}，j_0，j_{16}，j_{18}] 外，其余所有关节点
[ankle joints]	[j_{10}，j_{13}]
[heel joints]	[j_{11}，j_{14}]

下文列举出几个数据选择和处理的典型：

4.3.1 不经过任何预处理，特征提取

不经过任何数据预处理，输入特征为 [x，y]，[heel joints] / [x，y]，[ankle joints]，在固定机位的所有组的测试集表现上正确率平均在 92%，但是在跟随机位的所有组的测试集表现上正确率平均只有 65%。这意味着这个模型对于任何运动场景的泛化能力都比较差，而且如果一旦所选的这两个关节点信息因为遮挡有大段缺失时，模型就失去了推断的能力。

4.3.2 只进行归一化，特征提取

表 3 不同关节点的准确率

[10 joints]	55%
[8joints]	57%
[left/right joints]	49%
[ankle joints]	63%
[heel joints]	63%

输入特征为 [x，y]，[10 joints] / [8 joints] / [left/right joints] / [ankle/heel joints]，采取均值归一化策略，训练出来的模型，虽然在固定机位所有组训练得到的平均正确率区间在 [79%—91%]，但是在跟随机位所有组的表现能力不佳，平均正确率见表 3。

其正确率甚至低于逻辑回归的分类正确率，故舍弃这个方法。

4.3.3 部分降噪，归一化、特征提取、删除 x 值

输入特征为 [x，y] / [y]，[20 joints]，插值补全空值，采取均值归一化策略，训练出来的两个模型中，包含 [x，y] 的两个值的模型在人物出现画面左右移动的模型里，正确率只有 40%，即使是在固定机位纵深运动中人物也会有左右偏移，模型的精确率相比之前降低了 20%。但是这些问题都伴随 [x] 值被删除得以改善，那些正确率只有 40% 模型，在删除 [x] 输入后重新训练，正确率回到了 75%，而固定机位纵深运动中由于人物带来 20% 下降也被消除。故在本阶段研究中将 [x] 值删除。

4.3.4 完全降噪，归一化、特征提取、删除 x 值

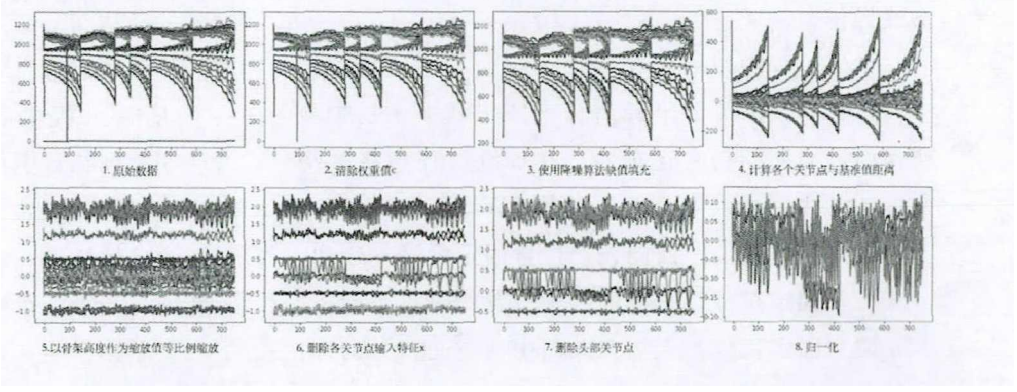

图 7

考虑到之前所有的训练测试都没有去除人物在画面中所在位置对数据带来的影响，因此才加入了新的降噪方法（见 3.5.2 的第二个降噪方法）。输入特征为 [y]，[20 joints]，插值补全空值，采取均值归一化策略，训练出来的模型对于模型的正确率提高了 10%左右。

图 7 展示了经过各步数据预处理方法数据精度提高的变化过程，顺序为从左至右，从上至下，每个图标纵坐标为进经过每一步精度收敛后的值，横坐标为时间（帧）。各图中不同颜色的曲线代表不同特征值的变化波动范围。

4.4 训练结果以及精确度

最终方案（参照上文 4.3.4）在不同的学习速率/学习权重值下，训练得到的结果除去人物沿纵深位置人物运动的情况正确率只有 70%～80%以外，在别的典型案例中，模型检测到正确的脚步运动周期准确率都在 90%左右。

笔者将已标注的数据集分成两类：一类叫做训练集，另一类叫做测试集。训练集用于 MLP 与 SVM 模型的训练，测试集被用于评判模型拟合的效果，得出正确率的值可以作为预处理方法和模型鲁棒性优劣的参考。其结果如图 8 所示：

	固定机位 人物位置固定的脚步运动	固定机位 人物向纵深位置的脚步运动	固定机位 人物在画面中从左向右/从右向左移动	机器跟随 人物人物正面向前移动	机器跟随 人物人物背面向前移动	机器跟随 人物人物侧面向前移动	平均值
SVM C=0.1	50.65%	52.12%	48.84%	52.77%	51.52%	51.17%	51.18%
SVM C=1	89.73%	58.22%	48.97%	81.08%	89.49%	87.77%	75.88%
SVM C=10	89.55%	77.19%	87.53%	87.20%	90.30%	87.49%	86.54%
SVM C=100	90.45%	81.56%	84.19%	89.19%	91.52%	87.37%	87.38%
SVM C=200	90.37%	79.97%	85.48%	88.05%	91.92%	87.88%	87.28%
MLP[20,20,10] a=0.001	89.61%	73.34%	80.85%	80.09%	93.74%	89.33%	84.49%
MLP[20,20,10] a=0.01	91.19%	73.87%	78.66%	80.51%	92.32%	86.15%	83.78%
MLP[20,20,10] a=0.1	89.53%	78.91%	83.42%	86.34%	92.73%	88.99%	86.65%
MLP[20,20,10] a=0.5	91.07%	80.90%	78.41%	79.23%	92.73%	88.99%	85.22%

图 8

可以发现，不同的学习权重值 c（SVM）和不同的学习速率 a（MLP）训练出的模型会对结果造成比较大的影响。SVM 模型在学习权重值 c 为 100 时训练完成的模型，对于不同状态运动的综合表现最好。MLP 模型在学习权重值 a 为 0.1 时训练完成的模型，对于不同状态运动的综合表现最好。

5 部署思路

5.1 脚步片段素材准备

在本次研究中，尚未尝试使用自动的方法获得脚步素材的片段，所以需要人工先从素材库中找出与画面中地面材质、鞋子材质符合的脚步素材，并将其切片分割为左、右两类的素材，对素材文件进行一致性命名，同种材质单只脚保留 20 个样本。

5.2 利用模型输出推断数据的结果自动编写 XML 序列

模型推断得出的数据结果分别为 0 和 1，分别代表左脚周期与右脚周期，通过编写一个脚本，使得在左右周期切换时标注并记录对应的帧序列帧数，这可以得到左右脚对应落下时的对应帧。将帧数序列转换为以 nn（frame）/30（frame）s 的格式[③]，这样就可以得到一个包含左右脚交替落下的时间序列，将时间序列编码成 XML 序列，同时 XML 序列中还包含了脚步素材的路径，以及素材本身的部分元数据。

程序选择脚步素材的过程是随机过程，将单只脚素材映射为 0 到 19 的序列，当程序检测到需要左脚素材时，程序会使用 random 方法去从 0－19 中挑选出一个数值，数值对应的脚步素材的名字与各脚步起始时间点会被添加到 XML 文件中，往后右脚素材同理重复一遍次流程，如此往复。

5.3 XML 序列绑定脚步片段导入时间线

将 XML 序列导入到支持 XML 导入的工作站中，以 DaVinci Resovle 这个软件为例，软件会自动将 XML 中的元数据信息，从对应文件夹中找出对应的声音素材文件，自动排布在时间线上，完成后再利用 DaVinci 的媒体文件导出功能，将时间线导出为 aaf、omf 交换文件格式，就可以进入正常的工业流程了。

6 实验的局限与展望

本次实验的不足主要体现在以下三个方面：

首先，在判定的过程中，需要事先采用人工的方式，考虑人物不同运动状态以及摄影机运动的状

态，分别进行模型的训练才能保证模型用于预测时的精度，自动化和智能化程度尚待进一步提高。

其二，由于声音素材的现状，智能化元数据标注还非常不完善，所以在实际应用环节上还需要通过手动指派素材类型的 XML 文件，很大程度上失去了大规模工业化的实用性。

第三，由于目前使用的神经网络相对比较简单，自发提取骨架特征难以实现，也从一定程度上影响了这个方法的大规模部署。

因此，在后续的研究中，可能会考虑在以下几个方面进行优化和改进：

首先，更换神经网络模型，大幅度提升性能，如使用 GCN 网络使得计算机能“读懂”骨架每一部分的含义，以及各部分对脚步落下影响大小的权重值，这样能大幅提升模型的鲁棒性以及精确度，当骨架某些部分被遮蔽时，计算机也可以依靠别的部分去推断脚步落下的状态。另一方面，GCN 网络还能根据骨架进行别的动作的判定。

第二，对于画面中识别对象长时间消失（如某一只脚）而造成的数据缺失，因人的步伐频率相对固定，除了采用前一个方法外，其实还可以尝试只使用一只脚作为依据，直接机算另一周期所包含的帧。

最后，也是最有价值的一点是，使用图像语义分割技术将脚部与地面的画面提取出来，再使用图像分类技术识别出地板的材质以及脚部鞋子的材质，进而根据分类得到的文字直接在素材库寻找对应的素材。❖

注释

①ImageNet：A large-scale hierarchical image database. ImageNet：一个大规模层级的图像数据库 Jia DengSocher，Li Fei-Fei，Wei Dong，Kai Li and Li-Jia LiR. Miami，FL，USA：IEEE Computer Society，2009.［C］2009 IEEE Conference on Computer Vision and Pattern Recognition（CVPR）. pp. 248－255.

②Scikit-learn：Machine Learning in Python. Pedregosaand Varoquaux，开源项目 G. and Gramfort，A. and Michel，V. and Thirion，B. and Grisel，O. and Blondel，M. and Prettenhofer，P. and Weiss，R. and Dubourg，V. and Vanderplas，J. and Passos，A. and Cournapeau，D. and Brucher，M. and Perrot，M. and Duchesnay，E. F. 2011，Journal of Machine Learning Research，pp. 2825－2830.

③nn 代表在整个序列的所在帧数，30 代表视频的帧速率，s 为单位（秒）。

参考文献

[1] Jia Deng，Richard Socher，Li Fei-Fei，etc. 2009 IEEE Conference on Computer Vision and Pattern Recognition［C］. Miami，FL，USA：IEEE，2009.

[2] 中国电影网．中影基地电影核心科技新突破——“中影·神思”人工智能图像处理系统［EB/OL］. https：//www. chinafilm. com/zydt/7077. jhtml，2018－12－17.

[3] OpenPose：realtime multi-person 2D pose estimation using Part Affinity Fields. SheikhCao and Gines Hidalgo and Tomas Simon and Shih-En Wei and YaserZhe.［J］. arXiv，2018.

[4] SheikhWei and Varun Ramakrishna and Takeo Kanade and YaserShih-En. Convolutional pose machines.［J］. CVPR. 2016.

[5] Scikit-learn：Machine Learning in Python. Pedregosaand Varoquaux，G. and Gramfort，A. and Michel，V. and Thirion，B. and Grisel，O. and Blondel，M. and Prettenhofer，P. and Weiss，R. and Dubourg，V. and Vanderplas，J.. and Passos，A. and Cournapeau，D. and Brucher，M. and Perrot，M. and Duchesnay，E. F. 2011，Journal of Machine Learning Research，pp. 2825－2830.

[6] Wikipedia. Support-vector machine［EB/OL］. https：//en. wikipedia. org/wiki/Support-vector _ machine，2019－5－5.

[7] Wikipedia. Multilayer. perceptron［EB/OL］. https：//en. wikipedia. org/wiki/Multilayer _ perceptron，2019－5－5.

[8] Openpose 开源项目［CP/OL］. https：//github. com/CMU-Perceptual-Computing-Lab/openpose.

[9] Support Vector Machines，支持向量机［DB/OL］. http：//www. support-vector-machines. org/.

[10] Multilayer perceptron，多层感知机 Multilayer Perception，Fuzzy Sets，and Classification. Sabjar J. Pal and Sushmita Mitra.［J］. IEEE，1992.

作者贡献声明：

郭境熙：论文框架设计，实验方案设计，数据处理，论文撰写，全文文字贡献 60%；

刘嘉：论文框架设计，实验方案优化，数据优化，论文修订，全文文字贡献 40%。

（下转第 19 页）

IMF Technological Development and Application in Cross-platform Distribution

◎Fang Jiexin, Cui Qiang (China Research Institute of Film Science & Technology)

Abstract: Under the trend of media content distribution and broadcasting cross-platform and media convergence, content distribution and broadcasting need to be carried out in different media formats for various terminals, which puts forward new requirements for the interoperability of content distribution, delivery and archive formats. In recent years, the Interoperable Master Format (IMF) has been used as a technical solution for the current cross-platform and media convergence distribution, and has formed a series of industry standards. This article describes the development, application and related technical details of IMF technology and standards, provides technical solutions for professional cinema, on-demand cinema, home channels, public media, mobile terminals and other media convergence distributions.

Key words: Interoperable Master Format; Cross-platform; Media convergence; Distribution; Archive

---

(上接第 12 页)

Exploring the Sound Post-production of Film and Television Based on Image Recognition and Neural Network Technology

◎Guo Jingxi, Liu Jia (Sound School of Beijing Film Academy)

Abstract: This study selects the most common footstep editing in actual work as the object, based on the open source Openpose framework to analyze and extract the movement posture of the character in the picture, and use two neural network learning methods Support Vector Machine (SVM) and Multilayer Perception (MLP) process the Openpose data separately, and train prediction models for different pictures and motion situations. In this process, the effects of different preprocessing combinations on model fitting were compared, and the results were finally encoded into a standard XML sequence compatible with a variety of audio workstations. Exploring an auxiliary method of sound post-production that is acceptable in terms of habit, accuracy, and practicality.

Key words: Sound editing; Image recognition; Neural network; Machine learning; Posture speculation