浙江大学

ZJUI 学院 2023 年暑期科研项目总结材料

中文论文题目 ： 基于计算机视觉技术的排球综合分析系统的搭建研究

英文论文题目： **An Integrated Volleyball Analyzing System Based on Computer Vision**

成员姓名：王晋文（3220111435）

李文韬（3220111412）

林子超（3220111892）

指导教师：王高昂

指导教师所在学院：ZJUI

暑期研究起止日期：2023 年 06 月 15 日-2023 年 07 月 15 日

摘要

我们计划做一个集动作识别、运动员再识别、赛后精彩集锦为一体的排球分析系统。经过前期调研，我们打算采用Segment Anything模型为比赛视频预处理，切割出运动员的准确轮廓；随后，用Multisports数据集训练出可以识别运动员动作的网络，进而用下游模块实现上述功能。在具体神经网络的选取上，我们重点调研了Resnet模型和Transformer模型。考虑到Transformer可以学习到全局特征，我们最终选择它作为我们的训练神经网络。经过数据集的预处理，我们将其整理成指定的目录，随后使用MMAction2网络对其进行训练，获得了较好的结果。我们后续的工作包括让Multisports的输入与Segment Anything的输出相适应、调研相应的下游模块、增加数据集的训练量等。

Abstract

We are planning to develop a volleyball analysis system that integrates action recognition, athlete re-identification, and highlight videoes. After preliminary research, we plan to use the *Segment Anything* model to preprocess the match videos and accurately segment the contours of the athletes. Subsequently, we will use the *Multisports* dataset to train a network that can recognize athlete actions, and then use downstream modules to achieve the above functions. In terms of specific neural network selection, we have focused on researching the Resnet model and the Transformer model. Considering that Transformer can learn global features, we finally chose it as our training neural network. After preprocessing the dataset, we organized it into a specified directory, and then used the MMAction2 network to train it, achieving good results. Our follow-up work includes adapting the input of Multisports to the output of Segment Anything, researching relevant downstream modules, and increasing the amount of dataset training.

1 前期调研与研发思路

为了使我们的设计贴合现实，我们对当下计算机视觉在排球领域应用的现状以及相关排球领域的数据集进行了调研。

1.1 计算机视觉在排球领域应用调研

不同于体操的AI打分已经用于奥运会这一现状，排球领域中计算机视觉的应用并不明显。通过进一步的查找资料，我们发现市面上的分析公司，如Hudl可以实现位于特定时间下的动作识别，如图1-1。由于其作用不够全面，识别精准度也有待提高。因此，我们希望搭建一个综合的排球分析系统，不仅在精准度上有显著提高，也进一步丰富其功能，实现实时对运动员的动作识别、比赛走向的分析、赛后精彩集锦等多项功能合一的排球分析系统搭建，这样可以满足排球运动员、教练、球赛观众等多方需求，实用价值较大。

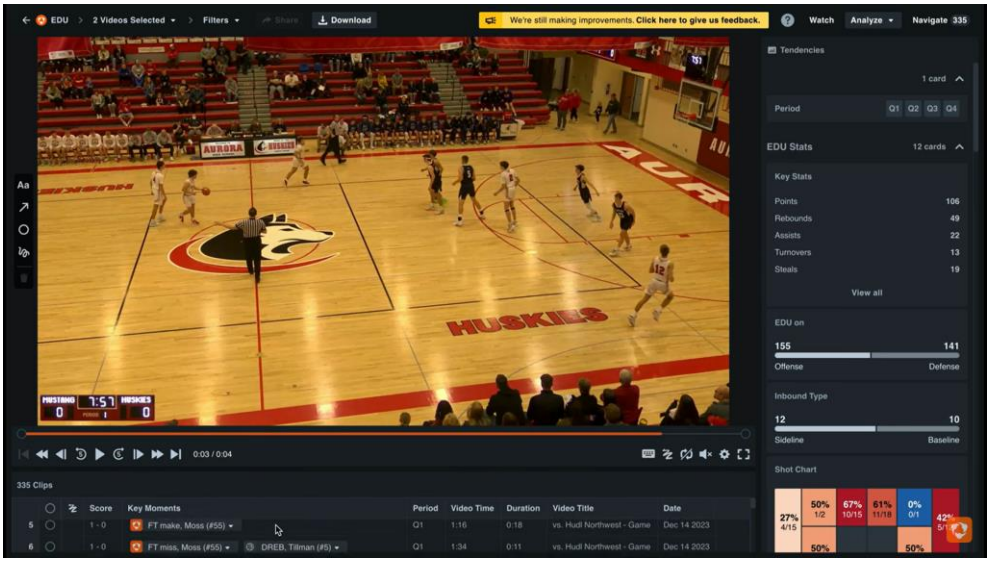

图 1-1 Hudl 识别演示

1.2 Segment Anything调研

在计算机视觉应用于比赛领域时，运动员识别精准度是一大难题。若用边界框识别，由于其包含背景内容，结果不够精确。Kirillov等人研发了Segment Anything模型，可以较为精确地切割出图片中物体的边框。其被应用于运动员识别的效果也较好（如图1-2）[1]。因此，我们打算使用该模型进行运动员的识别工作。

图 1-2 Segment Anything 演示

1.3 排球数据集调研

经过筛选，我们决定将Multisports作为我们的重点调研对象。Multisports是一个多人运动动作识别的数据集，其领域涉及篮球、排球、足球等多个领域。通过调研Multisports的相关数据，我们认为其可以作为我们的训练神经网络的数据集。Multisports为已标注的逐帧数据集，方便训练。同时，Multisports作为较新的数据集，采用MMAction2神经网络进行训练，其准确度也较高。另外，需要注意的一点是，Multisports使用边界框识别运动员，而这就要求我们在后续需要对其代码进行更改，使其适应Segment Anything的输出。

1.4 研发思路

基于 SAM 系统，我们可以自动化地将多人运动视频按照运动员分割成多个姿势序列，为多人运动数据的分析和理解提供了新的手段。

在该系统中，我们设计了一些使用者输入指令，用于根据用户需要自定义运动切割方式。首先，用户可以在系统中选择使用预设的运动切割方案，方便调用已有的分析系统。同时，使用者可以给定切割后的分类方法，如按照运动员的球衣号码进行分类，按照运动场上的位置进行分类等。随后，输入比赛视频，SAM 可以准确地切割各运动员的边框，识别出他们的位置。

在运动员切割完成后，系统基于SAM切割运动员的结果提供了多种可选的功能：动作识别及评估、运动员再识别、生成对特定运动员的报告与高光时刻集锦、

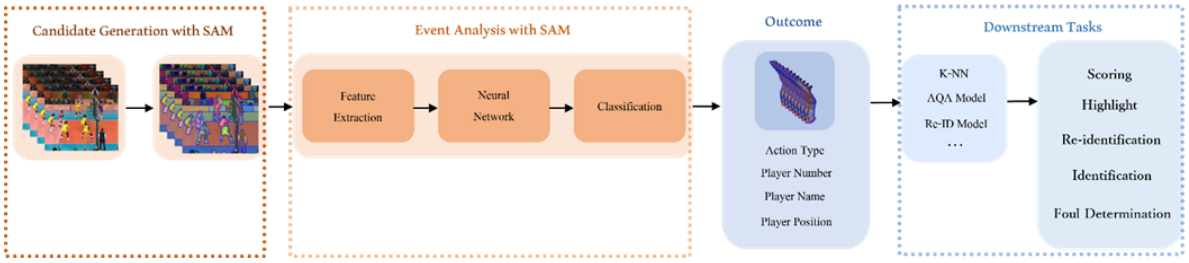

图 1-3 研发思路示意图

2 神经网络模型的选择

2.1 Resnet50模型

为了适应对比赛视频逐帧进行语义分割和识别分析的需求，我们对当前主流的神经网络进行调研和筛选，其中优势较为明显的有何凯明于2015年提出的Resnet50卷积神经网络。该网络的优势主要体现在其采用的残差连接方法可以有效解决大多数网络达到一定深度后的梯度消失问题，也就是避免由于网络层数太多而权重更新缓慢，训练效果不佳的问题。而该系统需要实现对网球比赛球员的高精确度跟踪和动作识别，利用Resnet50便能够搭建非常深的神经网络，提高图像识别的性能。[2]

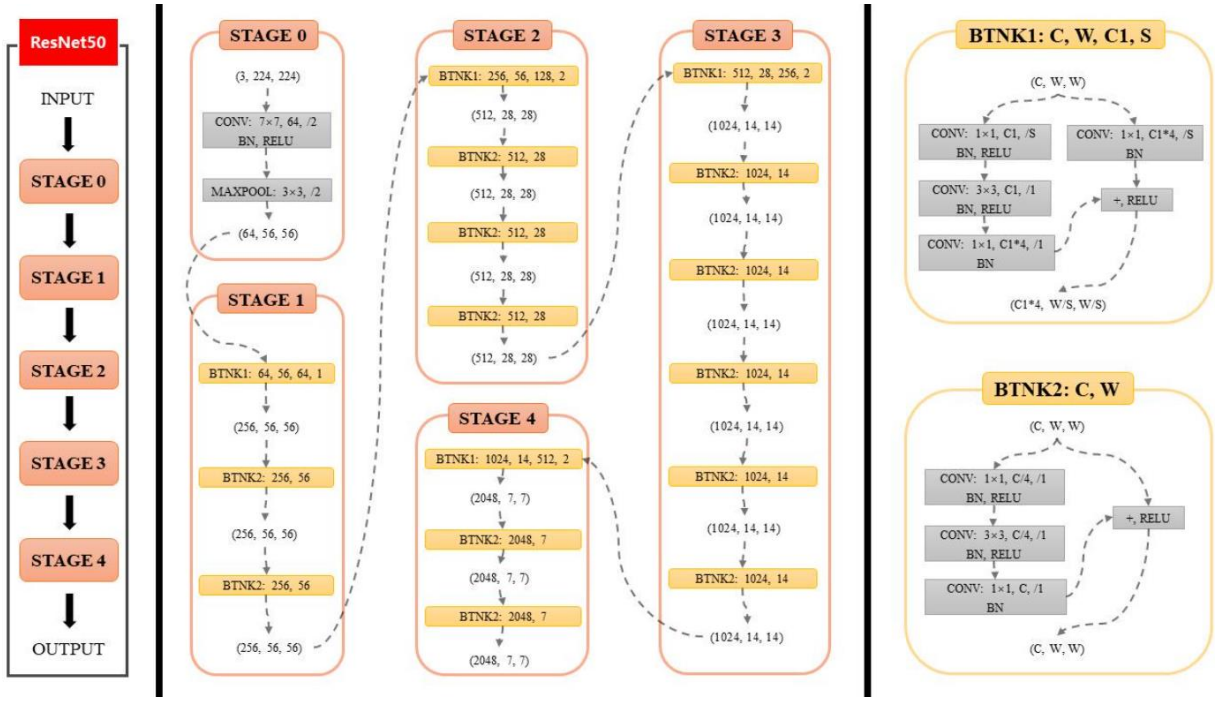

图 2-1 Resnet50 模型示意

2.2 Transformer

另一个同样值得注意的网络是 transformer。该网络摒弃了传统的 CNN 或 RNN 的模型，只采用注意力机制以实现识别局部与全局关系的功能，也就是可以识别局部序列中上下文的含义，可以准确提取出输入中的关键内容。且该网络模型可以并行计算，处理长序列基本不会丢失信息，与其他网络模型相比训练时间更短。该网络最初运用于翻译领域，训练后对文本的翻译准确度遥遥领先于传统网络模型。而在 2021 年 Alexey Dosovitskiy 和 Lucas Beyer 发表论文 VIT，将 transformer 模型首次运用于图像分类，经过训练后在分类任务中同样表现出色。[3]

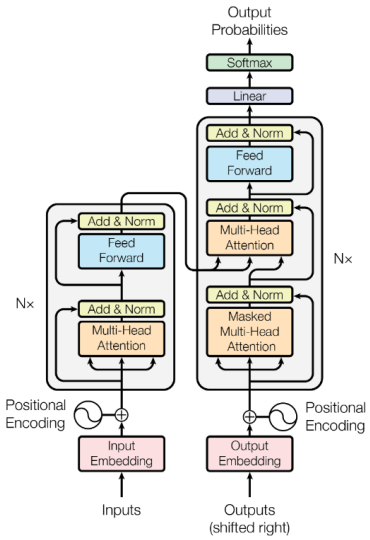

图 2-2 Transformer 模型示意

2.3 优劣对比及模型选择

我们最后选择了Transformer作为排球综合分析系统的主要模型。虽然我们使用的Multisports数据库使用的是Resnet网络进行训练，但经过对两个模型的优缺点进行比较分析后，我们认为Transformer更能适应任务的需求。如ResNet的卷积操作只能学习到局部的特征表示，而Transformer的自注意力机制可以学习到全局的特征表示，因而在分析比赛视频的过程中更不容易丢失主要信息，对于动作的打分或是比赛的评估等下游任务的准确度会更高。而Transformer模型需要较大的数据量进行训练才能够有效提高模型的预测准确率，因此后期还需要搜集尽可能多的排球运动的数据集以提升该模型性能。

3 数据集的处理与训练

3.1 数据集预处理

MultiSports由标注文件，人物候选框，和视频三类文件构成。每类文件均由训练集、测试集和验证集构成。我们采用了视频理解开源工具包MMAction2作为数据集的训练框架

[4]。MMAction2 基于 PyTorch 开发，可以方便地调用不同神经网络和预训练模型。在MMaction2中，常用的数据集和经过不同数据集预训练的神经网络可通过调用api的方式直接调用。对于每个具体的神经网络模型，训练参数可在其配置文件中修改，以起到方便修改训练策略的目的。MMaction2还提供了丰富的标注文件转换功能，通过MMaction2的处理脚本parse_anno.py，我们将标注文件转换成csv格式，将候选框脚本转换成dense_proposals格式，最后整理成以下文件：

```
mmaction2
├── mmaction
├── tools
├── configs
├── data
│   ├── multisports
│   │   ├── annotations
│   │   │   ├── multisports_dense_proposals_test.recall_96.13.pkl
│   │   │   ├── multisports_dense_proposals_train.recall_96.13.pkl
│   │   │   ├── multisports_dense_proposals_val.recall_96.13.pkl
│   │   │   ├── multisports_GT.pkl
│   │   │   ├── multisports_train.csv
│   │   │   ├── multisports_val.csv
│   │   ├── trainval
│   │   │   ├── aerobic_gymnastics
│   │   │   │   ├── v__wAgwttPYaQ_c001.mp4
│   │   │   │   ├── v__wAgwttPYaQ_c002.mp4
│   │   │   │   ├── ...
│   │   │   ├── basketball
│   │   │   │   ├── v_-6Os86HzwCs_c001.mp4
│   │   │   │   ├── v_-6Os86HzwCs_c002.mp4
│   │   │   │   ├── ...
│   │   │   ├── multisports_GT.pkl
│   │   │   ├──...
│   │   ├── test
│   │   │   ├── aerobic_gymnastics
│   │   │   │   ├── v_2KroSzspz-c_c001.mp4
│   │   │   │   ├── v_2KroSzspz-c_c002.mp4
│   │   │   │   ├── ...
```

图 3-1 Transformer 模型示意

3.2 数据集训练

整理好数据集后，在服务器上创建虚拟环境，并按照如下列表搭建环境：

```
System environment:
    sys.platform: linux
    Python: 3.9.0 (default, Nov 15 2020, 14:28:56) [GCC 7.3.0]
    CUDA available: True
    numpy_random_seed: 1503023888
    GPU 0,1,2,3,4,5,6,7: NVIDIA GeForce RTX 3090
    CUDA_HOME: /usr/local/cuda
    NVCC: Cuda compilation tools, release 11.3, V11.3.109
    GCC: gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0
    PyTorch: 1.12.1+cu113
    PyTorch compiling details: PyTorch built with:
  - GCC 9.3
  - C++ Version: 201402
  - Intel(R) Math Kernel Library Version 2020.0.0 Product Build 20191122 for Intel(R) 64 architecture applications
  - Intel(R) MKL-DNN v2.6.0 (Git Hash 52b5f107dd9cf10910aaa19cb47f3abf9b349815)
  - OpenMP 201511 (a.k.a. OpenMP 4.5)
  - LAPACK is enabled (usually provided by MKL)
  - NNPACK is enabled
  - CPU capability usage: AVX2
  - CUDA Runtime 11.3
```

图 3-2 参数配置

上传数据集,选用 Resnet50 网络和 kinetic400 预训练模型,按照训练集:测试集=1:9 进行训练，经过多轮训练，得到的最终准确率达 98%。

4 后续工作

4.1 修改Multisports的输入数据格式

在利用Multisports训练出可识别运动员动作的网络后，由于其利用边界框来识别运动员，而这与SAM输出的运动员的切割图并不相符。因此，为了实现更加精准的识别，我们将修改Multisports的输入部分，使其适应SAM对运动员的切割结果。

4.2 调研相应的下游模块

我们设想通过下游模块来实现对排球运动员的动作打分，对比赛的实时评估、对选手表现的分析等工作。为此，我们还需要进一步对下游模块进行调研，将Multisports输出的动作结果等各项数据进一步分析，得到最终的输出结果。

4.3 增加数据集训练量

如前文所述，我们采用Transformer作为神经网络模型，需要大量的数据集进行支撑。因此我们应进一步调研排球相关数据集。

结论

根据前期背景调研、神经网络调研结果、我们已经利用数据集Multisports训练出一个可以根据运动员的边界框识别出特定排球动作的网络，可以应用于后续的综合分析系统的制作。另外，我们也计划在后续的工作中，继续完成其它模块的制作，将SAM与该网络进行对接，寻找对应的下游模块等，从而将其整合成一个完整的网络。

引用

[1]He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep Residual Learning for Image Recognition* (arXiv:1512.03385). arXiv. http://arxiv.org/abs/1512.03385

[2]Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., & Girshick, R. (2023). *Segment Anything* (arXiv:2304.02643). arXiv. https://doi.org/10.48550/arXiv.2304.02643

[3]Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need* (arXiv:1706.03762). arXiv. http://arxiv.org/abs/1706.03762

[4] https://github.com/open-mmlab/mmaction2