**Heterogeneous Non-Local Fusion for Multimodal Activity Recognition**

1. **概述**

1. 这篇文章的动机是**利用模态融合的方法来提高活动识别的精度**，融合的信号源有视频信号和传感器信号

2. 其中**创新点是对Non-Local神经网络做出了一些创新，使得其能够对异构信号源进行融合**

3. 但是Non-Local神经网络原本是用来处理单个信号源，从而实现self-attention；所以当输入多源信号时，**面临着如何处理异构数据的挑战**

4. 其中涉及到的知识点有：

* 模态融合
* Non-Local神经网络
* 活动识别

1. **相关工作**
2. 从活动识别的角度

* 本文的RELATED WORK部分是从活动识别的角度进行阐述，分为基于视频信号和基于传感器信号的活动识别

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Video-based Activity Recognition | | **Sensor-based Activity Recognition** |
| Deep spatio-temporal networks | Two-stream networks [1, 2] | Single sensor input[8] |
| Long-term temporal convolutions [3] |
| Multiplier networks [4] |
| 3D convolutional networks | I3D [5] | Multimodal fusion[9] |
| Resnet3D [6] |
| Non-local Neural Networks | Cross-modal attention [7] |  |

表1

1. 将Local和Non-Local网络进行对比

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | **Local** | **Non-Local** |
| 特点 | 捕获局部特征 | 考量全局、捕获远程依赖 |
| 作用 | 绝大多数的网络是Local神经网络，所以作用范围很广，比如CNN可以进图像识别、RNN可以进行机器翻译等等 | 作用较为单一，主要是Attention学习，一般作为网络的一部分使用（而且多用于模型的底层），而不是单独工作的 |

表2

1. **Local神经网络**
2. Local神经网络的例子

* 在讲Non-Local之前，很有必要讲一下什么是Local神经网络，因为它们是相对而言的
* 举两个深度学习领域的典型例子： **CNN网络处理图片、RNN网络进行机器翻译**

1. CNN网络处理图片

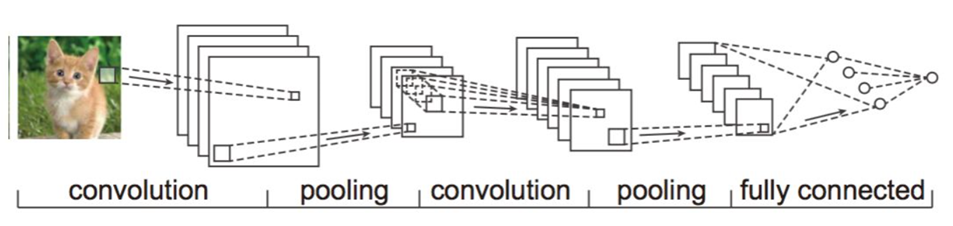


图1

* 从上图可以看到，在做卷积和池化等计算时，是对图片一个个局部进行计算（其实是为了提取有效的特征）；当filter是3x3的大小时，那么这时计算的就是图片中3x3的9个像素的大小
* 所以，**CNN网络为什么是Local的体现在不能一下子对整个图片进行计算，而是对图片每个局部进行特征提取**

1. RNN网络进行机器翻译

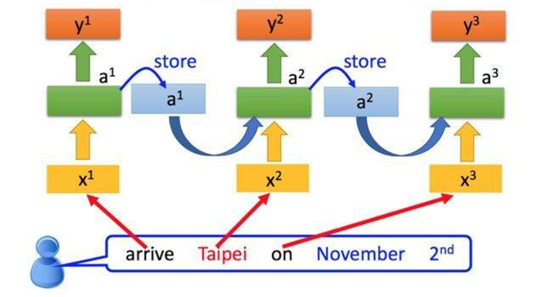


图2

* 从上图可以看到，机器翻译时输入的是一个一个单词，然后将上一个单词计算后的输出作为下一个的输入，来找到单词之间的联系（也叫远程依赖）
* 从这个处理过程可以看到，**RNN网络是一个一个单词的计算，而不是一下子计算整个语句，输出翻译结果，RNN网络属于Local神经网络体现在这里**

1. **Non-Local神经网络**
2. 什么是Non-Local神经网络？

* 了解了什么是Local的概念之后，Non-Local的概念其实就是相对的，也就是说，**它可以处理全局的信息**，而不是对局部进行处理

1. 用图片进行解释

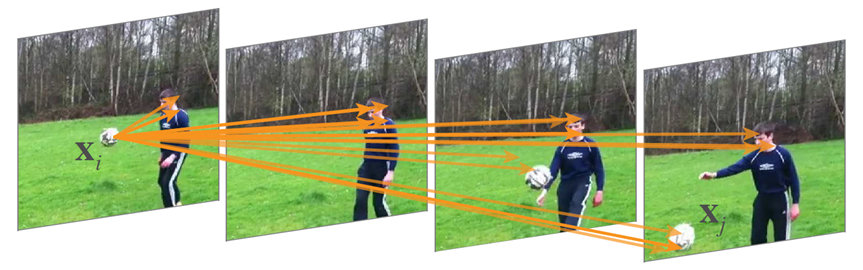
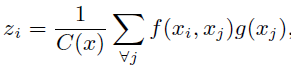


图3

* 做的工作是对视频信号进行处理，从而进行视频分类
* 首先，这是一个视频的四个帧，其中是第一帧中的一个位置（像素点）；会发现这个位置不仅关联了本图片的其他位置，还关联了其他帧的位置（图片中只画出了一部分关联性较强的位置），**这也是Non-Local的体现，因为这个网络是考量全局的**

1. 用公式进行解释  
   

* **变量代表的意义**：是输入信号，和在前面的图片上可以看到，是图片上两个不同的位置（可以是同一帧图片，也可以是不同帧图片），然后和下标其实是位置的index，可以简单理解为位置的编号；是输出结果
* **函数的意义**：函数用来计算位置和所有位置之间的关系，计算结果是一个标量（即常量，相对于矢量的一种说法），**Non-Local其实就体现在这里是所有位置**；g函数计算输入信号在j位置的representation，是一个简单的线性函数；其中C(x)是用来标准化的（就是控制输出的的值的大小，没有实际的意义）
* **特点**：这里其实能看出Non-Local网络的其中一个特点：**可以计算时空信息**，计算同一张图片的两个不同位置的联系时体现空间，计算不同图片（两帧之间）的两个不同位置时体现时间，可以同时计算所以体现了时空特性，所以Non-Local可以很好的应用于视频的信息计算
* **总结**：通过对公式的各部分进行分析，可以看到其实质是考量不同位置之间的关系；换句话说，若某个位置跟其他位置关系密切，那么说明这个位置是很重要的，我们从这个位置就能得出很多的信息，那么就可以赋予更大的权重（Enforced by representations from all other locations）

1. **Non-Local Block**
2. Non-Local Block其实是对Non-Local神经网络的一个封装
3. 如下图所示，会发现Non-Local Block的一个很重要的特点：输入的张量X与输出的张量Z维度是一样的，所以Non-Local Block可以很轻易的嵌入其他网络

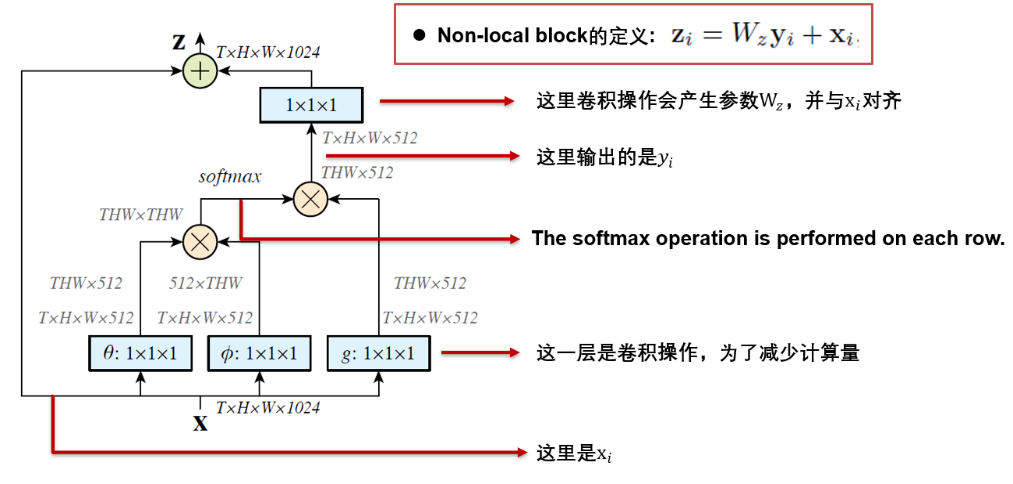
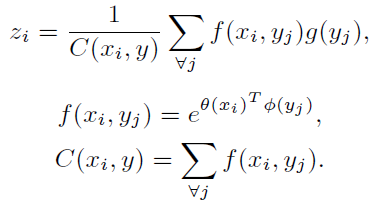


图4

1. **模态融合**
2. 前言

* 基于以上对Non-Local Block的了解，可以看到输入是同一个张量X，也就是只有一种信号源；那么如果这里输入的是不同的张量X和Y，就实现了模态的融合
* 当然，这需要解决一些问题，比如这里张量的维度不再一样了，怎么才能进行不同维度张量的计算

1. 用公式进行解释



* 这里把前面的换成即可，但是意义上也发生了一些改变
* **变量代表的意义**：x是第一个模态的信号，y是第二个模态的信号，i不再是位置的编号，而是信号x的dimensional index（维度的编号），j是信号y的dimensional index；是输出结果
* **函数的意义**：f函数用来计算x和y之间的关系，g函数计算输入信号x在y的representation，是一个简单的线性函数，同时也maps y into the same embedding space，意思就是调整维度大小（通过卷积从M调到K）；C(x)还是用来标准化的
* **关键点**：可以看到同时输入x和y的只有f函数，所以关键在f函数，文中指出**只需要将两个模态的输入信号的channel dimensions调至同样大小即可运算**，这也是这篇文章最主要的问题；比如x张量是𝐷1 × 𝐷2 × ... × 𝐷𝑝 × 𝑁，y张量是𝑆1 × 𝑆2 × ... × 𝑆𝑞 ×𝑀，只需要N和M的大小相同K即可
* **详解**：能计算的原因其实是利用了张量的特性；举个例子，二维张量是一个矩阵的形式，3X2大小的矩阵与2X4大小的矩阵能相乘的条件是第一个矩阵的列与第二个矩阵的行相同；所以张量相乘需要第一个张量最后一个维度与第二个张量第一个维度相同（如下图）
* **总结**：至此可以总结一下该block的所有特性
  + - 首先，可以计算时空信息，**捕获远程依赖**
    - 其次，输出维度与x维度一致，**可嵌入其他网络**
    - 最后，**其实际作用是通过y来加强x的学习**

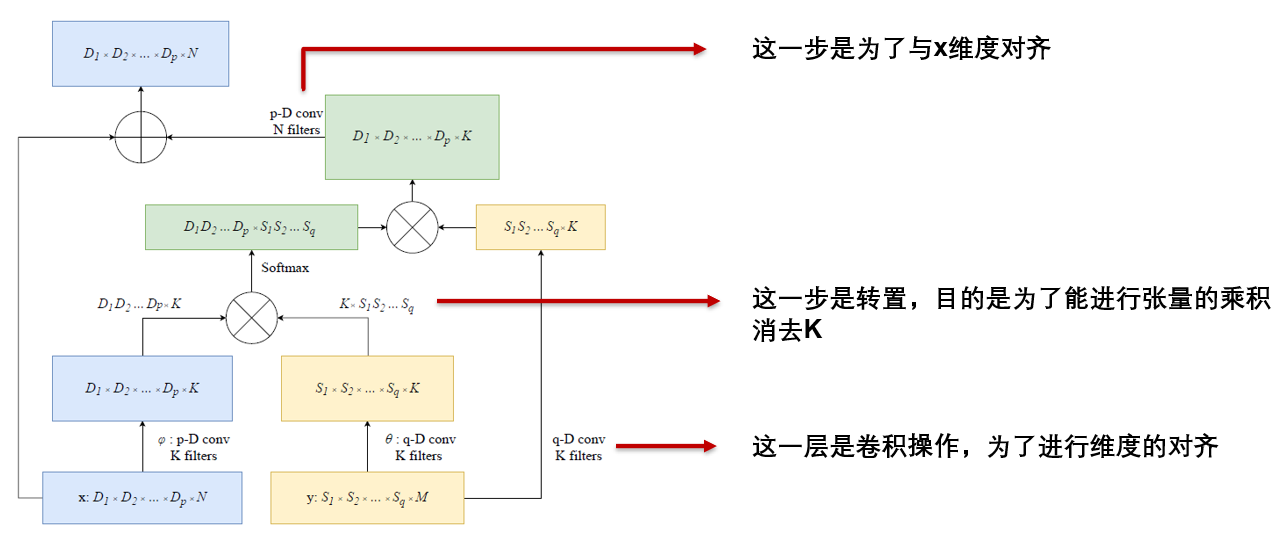


图5

1. **一些扩展**
2. 文中给出了三种不同的模型，实际想考量的分别是：以传感器为主要信息源，视频为辅的效果；以视频为主要信息源，传感器为辅的效果；对称模型的效果，也就是两种模态所起的作用是相同的

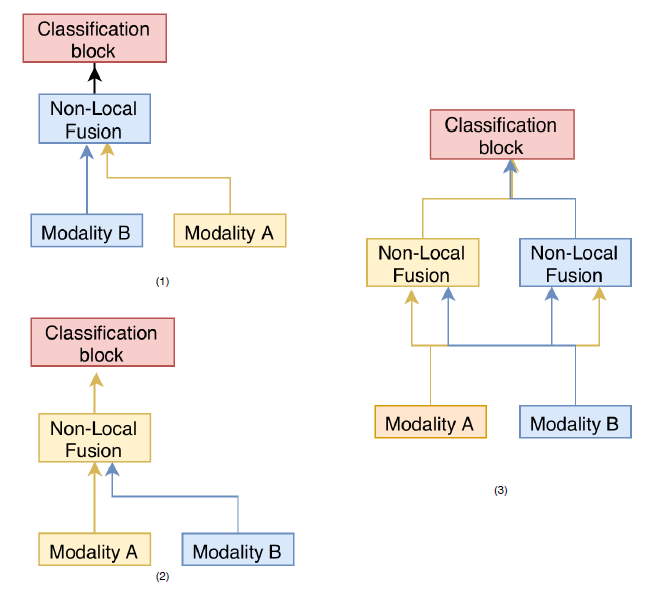


图6

1. **实验部分总结**
2. 实验设置

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | |  |
| 活动分类 | 围绕修电梯这个活动分为八个动作 | | neutral activity |
| pressing a button |
| unplugging the cables |
| plugging the cables |
| disassembling a button |
| assembling a button |
| using a screwdriver |
| using a hummer |
| 张量维度 | 传感器：134X18 | | 视频：79×224×224×3 |
| 人员设置 | 19个人 | | 每人进行16-20分钟的活动 |
| 硬件设置 | 手套（包含四种传感器、两只手都需要穿戴） | accelerometer | 视频  (使用Gopro HERO7胸部摄像机采集) |
| gyroscope |
| magnetometer |
| pressure  sensor |
| 模型训练 | 将19人的数据分为16个训练集、2个验证集、1个测试集 | | 学习率：0.001  指数加权平均值：0.9  批处理大小：8 |

表3

1. 评估内容

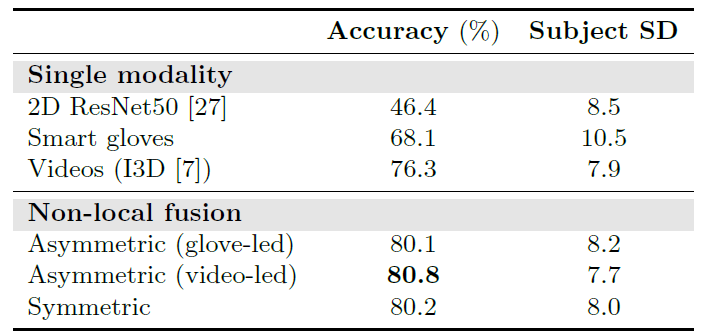
* 首先是对融合模型的分类准确率进行评估，对比的对象主要有单个输入的模型，融合模型又对比了以手套传感器数据为主、对称模型两种，**最终结论是以视频信号为主，传感器信号为辅的融合效果最好**

图7

* 然后为了了解融合模型相对于每个具体动作的识别有多少提高，进行了具体的量化分析

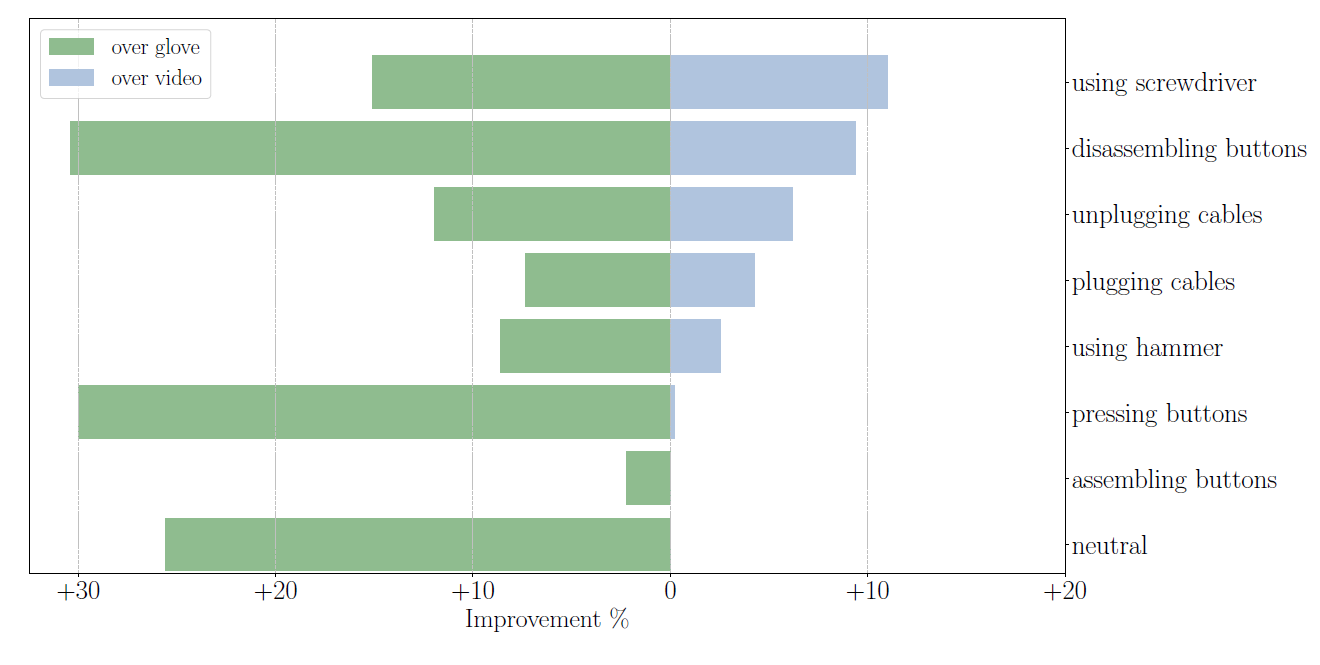
****

图8

* 对信道维度K的大小进行探索

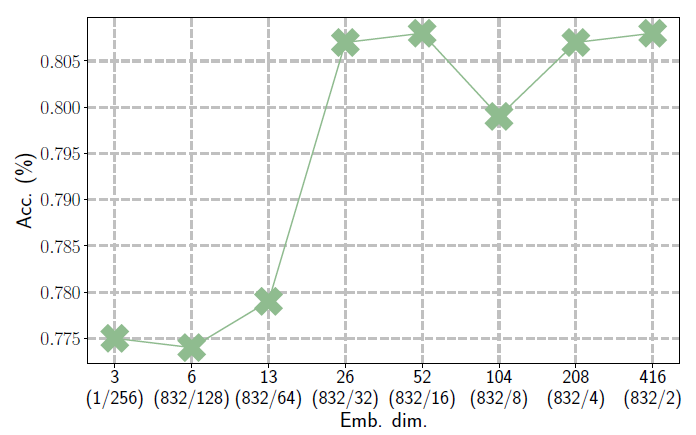


图9

* 为了探测手被遮挡时的识别精度，分别对三种情况进行分析：两只手都可见、一只手被遮挡、两只手被遮挡

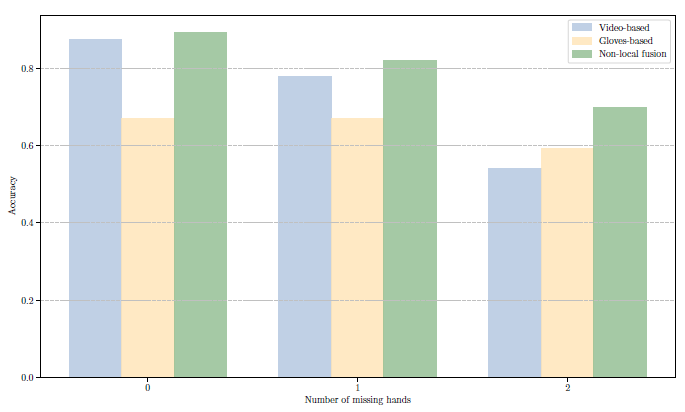


图10

1. **总结和启发**
2. 该模型不仅可以用于视频和传感器信号，还可以用于任何张量形式的输入，只需要通过卷积层进行信道维度的对齐即可
3. 为了减少计算量，可以考虑对模型进行一些改进

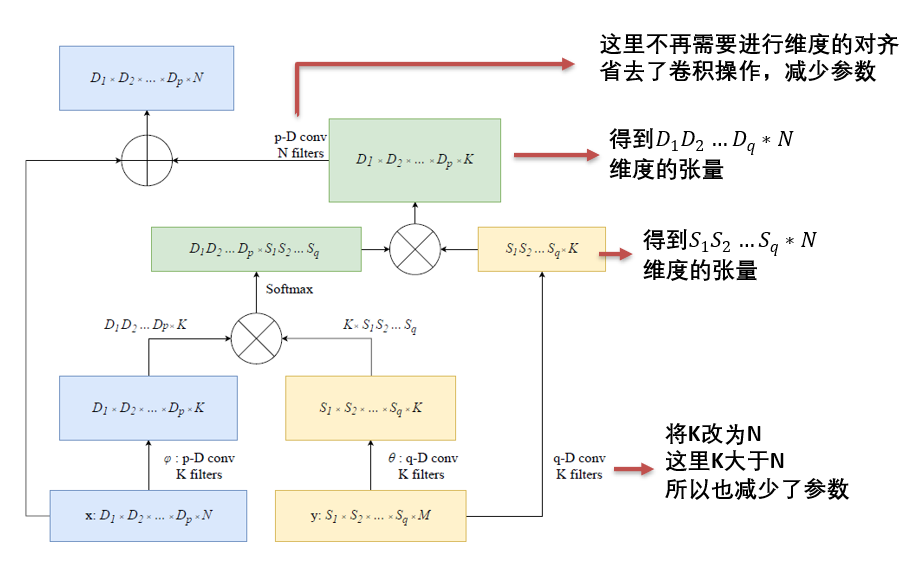


图11

**参考文献**

[1] C. Feichtenhofer, A. Pinz, and A. Zisserman. 2016. Convolutional Two-Stream Network Fusion for Video Action Recognition. InCVPR.

[2] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In Advances in neural information processing systems.

[3] Gul Varol, Ivan Laptev, and Cordelia Schmid. 2017. Long-term temporal convolutions for action recognition. IEEE transactions on pattern analysis and machine intelligence 40, 6 (2017), 1510–1517.

[4] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. 2017. Spatiotemporal multiplier networks for video action recognition. In CVPR. 4768–4777.

[5] Joao Carreira and Andrew Zisserman. 2017. Quo Vadis, Action Recognition A New Model and the Kinetics Dataset. CVPR(2017).

[6] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. 2017. Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet CoRR (2017).

[7] Lu Chi, Guiyu Tian, Yadong Mu, and Qi Tian. 2019. Two-Stream Video Classification with Cross-Modality Attention. In ICCVw.

[8] Korbinian Frank, Mara Jos Vera, Patrick Robertson, and Tom Pfeifer. 2010. Bayesian Recognition of Motion Related Activities with Inertial Sensors. In UbiComp’10.

[9] Alexandros Andre Chaaraoui, Jos′e Ram′on Padilla-L′opez, Pau Climent-P′erez, and Francisco Fl′orez-Revuelta. 2014. Evolutionary joint selection to improve human action recognition with RGB-D devices. Expert systems with applications 41, 3 (2014), 786–794.