人工智能视觉:具有物体检测自定义技能和先进的语音交互能力的智能音箱的设计和实现

**摘要**：亚马逊提供了一个基于云的开发者控制台，用于编写和部署自定义技能，这些技能可以与运行在主机硬件上的Alexa语音服务设备SDK整合。当这个部署的自定义技能被使用特定的语音请求调用时，主机硬件会执行所请求的任务并发回其响应。目前，开发者已经在现有的Alexa设备上实现了人工智能算法，作为其定制技能的一部分，其重点是基于语音的应用，而不是基于实时图像或视频的应用，因为目前的Alexa设备没有摄像头。因此，有必要设计一个先进的支持摄像头的Alexa智能音箱平台，并将其提供给开源社区，以促进实施基于图像/视频的人工智能算法，作为他们定制技能的一部分。本文描述了一个最先进的支持摄像头的、基于Linux的现代Alexa智能音箱原型的设计和开发。一块芯片拥有先进DSP的麦克风阵列与原型机连接、为Alexa语音服务提供经过语音算法处理的语音数据，以实现无缝的﹑全双工的用户与Alexa互动。最后，基于深度学习和计算机视觉的物体检测自定义技能在开发的原型上进行实现和测试。

**关键词**：Alexa语音服务，智能扬声器设计，麦克风阵列，语音算法，深度学习，单次检测器，Open CV

# 引言

半导体技术的进步在提高芯片组的性能和容量的同时，也重新降低了尺寸和成本。此外，人工智能框架和库的进步带来了在消费物联网设备的资源有限的边缘容纳更多人工智能的可能性。这些进步使消费类电子设备(如智能音箱)有可能被制造成小尺寸，但能够运行强大的算法来捕捉进程和理解语音命令。这些设备对我们的日常生活产生了巨大的影响[1]。如今，传感器是我们环境中不可或缺的一部分，它提供了连续的数据流以建立智能应用。一个例子是具有多个互连设备的智能家居场景。在智能家居场景中，多个智能设备（如智能安全摄像机、视频门铃﹑智能插头﹑智能碳氧监测器﹑智能门锁和火灾报警器等)相互连接，并相互协作，为一个共同的目标服务。智能音箱是此类智能设备中的一种，正在被普通用户广泛采用，并成为智能家居的一个组成部分。人工智能助手是基于最新的、可以理解基于语音的命令，并控制智能家居的复杂集成系统的智能音箱。用户通过呼唤“Alexa”唤醒词来唤醒Alexa人工智能语音助手，开始与普通的Alexa智能音箱进行互动，随后进行常规的对话式互动。

目前，开发人员已经实现了人工智能算法，重点是提高对话式人工智能系统的性能[2]。在智能音箱上实现实时图像/视频的人工智能算法的工作非常有限，或者说没有，因为目前的智能音箱设备没有摄像头。因此，有必要设计一个先进的支持摄像头的Alexa智能音箱平台，并将其提供给开源社区，以促进在智能音箱上实现基于图像/视频的AI算法。本文描述了设计和开发一个最先进的支持摄像头的﹑基于Linux的现代Alexa智能音箱原型。本文讨论的现代Alexa智能音箱是使用现成的硬件组件（Raspberry Pi、ReSpeaker v2、Raspberry Pi相机、普通扬声器)组建的，在这项工作中，为了提供一个无缝的﹑全双工的用户与Alexa的互动，我们选择了一个带有板载芯片的麦克风阵列，它承载了基于DSP的语音算法，并用于捕捉﹑处理和提供给Alexa的噪音抑制的语音反馈。因此，我们的概念验证原型展示了丰富的用户体验，通过提供改进的语音互动设备与智能音箱进行互动。最新的相关工作是由来自[3][4]的作者完成的。他们使用相同的ReSpeaker v2麦克风阵列，为他们的麦克风阵列和基于语音算法的智能助听器原型提供先进的语音交互能力，在他们的工作中，麦克风阵列被用来捕捉和处理来自各个方向的声音，通过抑制不需要的噪音，包括其他人的讲话，智能地确定优先次序并提供主要发言人的声音，麦克风阵列上的语音算法被他们用来自动选择和交替使用发言者的声音。而在我们的工作中，麦克风阵列被用来捕捉﹑处理并向Alexa提供经过噪声抑制的语音信号，以实现无缝、全双工的用户-Alexa互动。

# 拥有检测自定义技能和高级语音交互能力的智能扬声器概述

这项工作的主要目的是使智能扬声器原型在用户要求时进行物体检测。为了实现这一目标，如图1所示，在智能音箱原型中加入了一个摄像头模块，为了启动这个物体检测的自定义技能，用户必须呼唤Alexa，"问星期五她看到了什么"。当使用这个语音命令时，第2.5、2.6和2.7节中的脚本就会出现。2.5、2.6和2.7部分的脚本执行它们的任务，并向用户提供智能扬声器检测到的物体的名称。第二个目标是捕捉并向Alexa提供高质量的噪声抑制语音输入，以实现无缝的﹑全双工的用户-Alexa语音交互，为了实现这一目标，这里使用了ReSpeaker v2麦克风阵列，而不是单一的麦克风，因为它可以将语音与唤音分开。此外，这个麦克风阵列有一个内置的高性能处理器，加载了基于语音算法的高级数字信号必理(DSP) 芯片，使用户能够在离智能音箱5米或更远的地方与Alexa互动，在房间里走动时进行互动等。该麦克风阵列的作用和使用它为Alexa捕捉、处理和提供语音输人的好处将在2.2节进行讨论。第三个目标是改善用户体验，确保智能音箱在没有叫出唤醒词时不会被意外激活，同时确保A1exa的唤醒词能从输入的音频流中高精度地被发现。为了做到这一点，在第2.4节中讨论的第三方唤醒词引擎被整合到A1exa系统中。2.4节中讨论的第三方唤醒词引擎与Alexa语音服务C++SDK集成在2.3节一起。

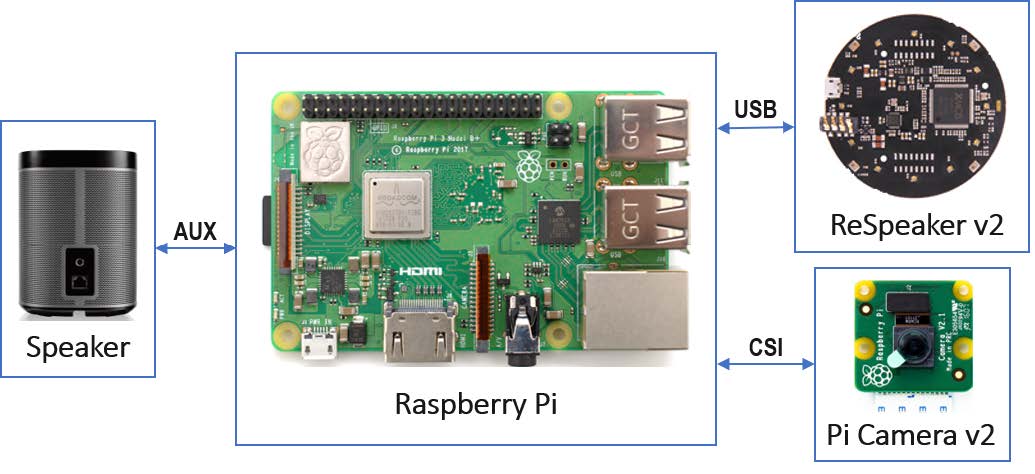


图 1 智能扬声器的硬件原型

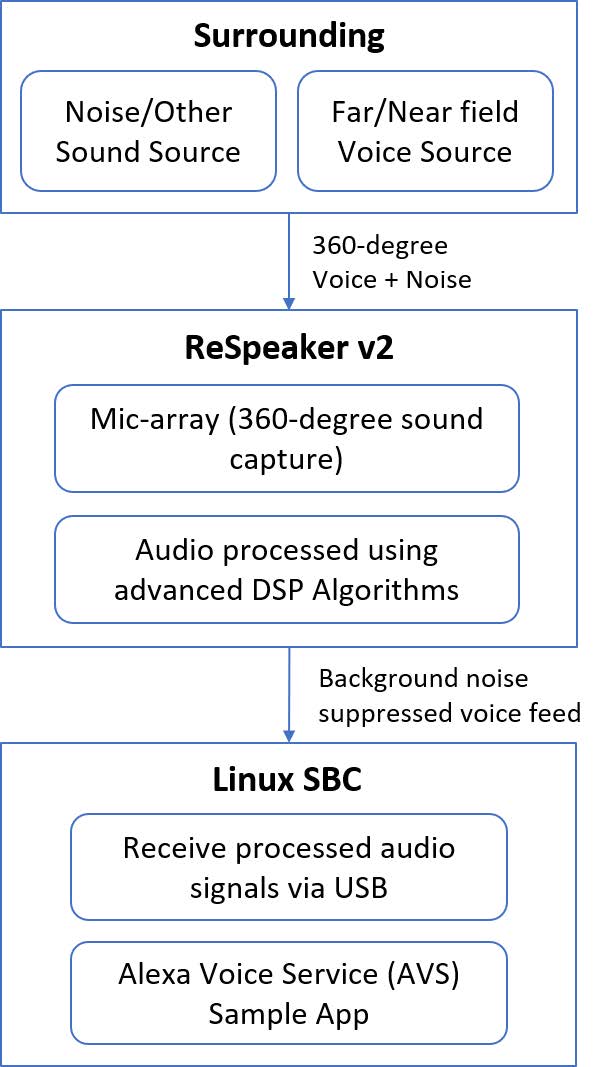


图 2 流程图，说明麦克风阵列的作用，捕获并向Alexa语音服务提供经过语音算法处理的语音资料

## 2.1智能扬声器原型的硬件组件

如图1所示，这个现代智能扬声器的原型是使用带有内DSP的现成高级麦克风(ReSpeakerv2)、相机模块(Raspberry Pi相机)和一个与单板计算机连接的普通扬声器建造的。

a.选择单板计算机（SBC）：BeagleBone Black, Orange Pi 3，LattePanda 2G/32G和Banana M4是我们感兴趣的SBC。选择Raspberry Pi 3b+型(Pi4尚未发布)是考虑到它的外形、价格、性能平衡、低功耗、与现成设备的兼容性﹑社区创建的指南教程和支持。利用外部库编写的Python脚本被部署在Raspberry Pi Linux SBC上。

b.选择相机单元：对于实时计算机视觉应用，首选Raspberry Pi Camera V2，因为它能够实现1080p 30fps的视频编码和500万像素的静态照片质量。由于相机通过CSI连接器直接连接到GPU，如图1所示，所以对Pi的CPU只有一点影响，使其可用于其他必理。大多数具有成本效益的网络摄像机都没有像Pi摄像机那样的内置编码。因此，网络摄像机使用额外的CPU资源，导致系统的整体性能下降。

c.麦克风阵列的选择：麦克风是智能音箱系统的重要组成部分。由于我们需要使用语音算法对声音进行预必理，所以重点是内置先进的DSP算法的麦克风阵列。ReSpeaker v2、Matrix Creator、PS3 eye、Conexant4-mic开发套件﹑MiniDSP UMA-8、Mi-crosemiAcuEdge ZLK38AVs是我们感兴趣的麦克风阵列。当距离增加时，ReSpeaker v2对热词检测有很好的成功率，并在无声房、有白噪声的房间和有背景音乐的房间进行测试[5]。PS3Eye比ReSpeaker v2有优势，但本项目选择ReSpeaker v2是因为带CSI接口的Raspberry Pi相机比PS3Eye相机对OpenCV环境的支持更好；第二个原因是，ReSpeaker有一个由12个RGB LED组成的像素环，除了扬声器单元之外，还可以用于视觉反馈。

## 2.2 基于语音算法的先进的语音交互能力的麦克风阵列

XVF-3000芯片上的固件(存在于Re-Speakerv2硬件上）通过USB向Linux系统产生六个通道的麦克风输出。通道零包含使用高级DSP算法处理的音频。通道一至四包含来自与通道编号相对应的麦克风的原始数据。通道五提供原始音频，是ReSpeaker v2上四个麦克风的所有原始音频信号的组合。图2显示了这个麦克风阵列的作用。这里，来自通道0的音频信号被用于唤醒单词的发现，同时也作为语音输入到Alexa。使用ReSpeaker v2和Alexa的好处如下。

a. 远场语音捕捉：可以在五米或更远的距离上捕捉和处理原始的微型电话输人，唤醒并与智能音箱互动；

b. USB音频类1.0 (UAC1.0)：USB音频用于将数字音乐从Raspberry Pi发送到ReSpeaker v2上内置的数模转换器(DAC)。1.0类最多可以发送24位/96kHz的高解析度文件。通过利用这一点，我们可以绕过Raspberry Pi的内部声卡，让USB DAC以更好的质量播放来自Alexa的音频响应。

c. 12个可编程的RGB LED像素环。ReSpeaker v2上的RGB LED像素环用于直观地指出语音信号到达的方向（来源)。像素环库用于通过USB接口处理LED像素，根据主程序的要求改变颜色和亮度。

d. ReSpeaker V2上的数字信号处理算法。

i. 波束成形。所有MEMS麦克风都具有全向性的拾音响应，这意味着，它们对来自传声器周围任何地方的声音的响应都是一样的。定向响应或波束模式可以通过在一个阵列中配置多个麦克风来形成。因此，使我们能够检测和跟踪智能音箱用户的声音在整个房间的位置。当智能音箱用户与智能音箱互动并在房间里走动时，麦克风光束的闰度会自动调整以跟踪他们的声音。因此，可以有效地指向用户的方向并抑制来自其他方向的噪音或混响信号。

ii.噪声抑制：在声学波束成形中，传声器阵列中的传声器的空间关系实现了主动的传声器噪声抑制和控制。如果声源相对于麦克风阵列的方向是已知的，那么可以设计一个声学波束成形器来通过来自感兴趣的声源的信号，并过滤掉从其他不同方向接收的声音信号。这种麦克风阵列降噪的方法最适用于在多人交谈时需要听到一个人的声音的情况。噪声抑制消除了静止的（点噪声)和非静止的背景声音。

iii.消除混响：在任何房间里，一个人的声音会在房间周围的硬表面上产生混响（反射），例如窗户或电视屏幕。去混响可以消除这些反射，清理语音信号。

iv.声学回声消除：在与电子设备互动时，在某些情况下，用户会听到他们的声音（有时会有明显的延迟）。这种体验被称为声学回声。控制和消除声学回声对基于语音的系统（如智能扬声器）至关重要。例如，如果智能音箱用户在电视上以最小的音量观看电影，并同时向智能音箱提供语音输人，现在麦克风将同时捕获用户的声音和电影的声音(声音回声)。这种声学回声会从语音输入中被消除，这样就可以更准确地从捕获的音频中提取文本。

## 2.3 Alexa语音服务C++ SDK

在设置智能音箱硬件和部署脚本的同时，Pi必须在亚马逊开发者控制台注册为一个设备，并创建一个安全配置文件。遵循云端设置的详细步骤说明[6]，并提供生成config.json的路径，同时从其SDK构建Alexa AVS样本应用程序。这个基于C++库的AVS设备SDK使我们能够将Alexa集成到智能音箱的原型中。智能音箱与AVs的互动是通过这个Alexa样本应用程序进行的，这个应用程序是通过官方SDK为Raspberry Pi建立的。在使用Alexa AVS C++SDK之前，我们用Raspberry Pi和ReSpeaker v2测试了Python版本的Alexa语音服务应用程序[7]。观察到了以下结果；

a.在与Alexa互动了一段时间后，Alexa的声音变得模糊和低沉。只有在重新启动Pi之后，这个问题才能得到解决。

b.在发现Alexa唤醒词后，会有一个最小时间的延迟（大约0.5秒)，之后音频会流向Alexa云。正如在[6]和图3中提到的，多个组件组成了C++ AVS SDK，音频数据在其中流动。最初，信号处理算法被应用于输入和输出音频通道，以产生经过处理的清晰音频。如果来自ReSpeaker的四个麦克风的原始音频数据被作为输入提供，那么这个第三方音频信号处理器就会结合并提供一个单一的音频流给架构中的下一个组件。但在这里，我们已经提供了一个单通道音频流，由ReSpeaker v2的XVF-3000芯片上的DSP处理。架构的其余子部分按照[6]和图3所述执行其功能。KITT.ai的Snowboy和Sensory[8]是两个第三方唤醒词引擎，其中任何一个都必须是SDK构建的一部分，以便从输入流中发现Alexa的唤醒词，从而提供免提交互。这两个引擎都用这个基于ReSpeaker v2的Alexa智能音箱设置进行了测试。Snowboy，唤醒引擎被选中，并被用作构建Alexa AVS样本应用程序的插件，因为它只消耗Raspberry Pi CPU的8%以下，并且在唤醒词检测方面有更好的成绩。

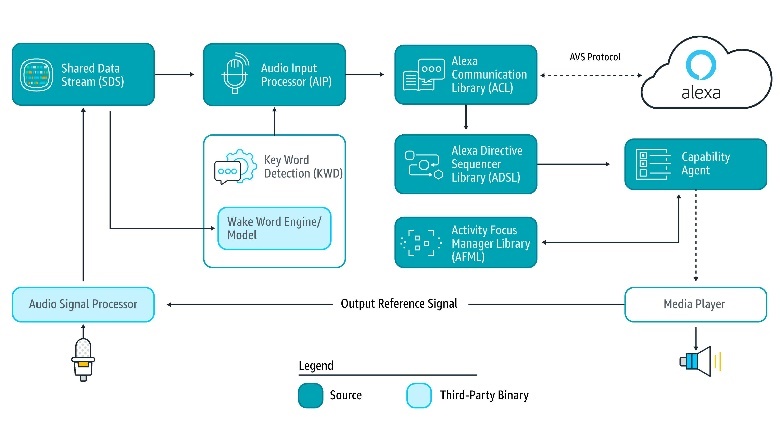


图 3 AVS Device C++ SDK组件之间的数据流

## 2.4 Snowboy唤醒引/擎发现Alexa的唤醒词

Snowboy引擎确保智能扬声器在没有叫出唤醒词的情况下不会被意外地激活。唤醒词检测引擎的准确性是通过绘制每小时假警报（有很多误报）与漏检率（引擎错误拒绝的唤醒词语料的百分比）来衡量的。四个不同的唤醒词检测引擎的ROC曲线显示在图4中。在这里，Snowboy唤醒词引擎的漏检率最低，它比其他引擎更准确。将唤醒词引擎与基于语音的人工智能助手和智能扬声器集成的原因[9]。

a.隐私：麦克风不一定要一直听；

b.成本:当数据一直流向云端时，是不切实际和昂贵的；

c．消耗功率。语音助手在智能手机﹑可穿戴设备和智能音箱上运行，预计待机时间最长；

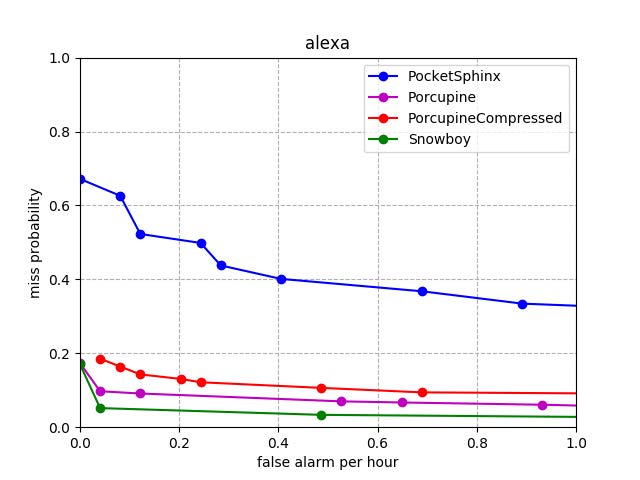


图 4 流行唤醒词引擎的ROC曲线

## 2.5基于深度学习和Open CV的物体检测

当用户呼唤Alexa，"问星期五她看到了什么"时，目标是通过利用MobileNets和Single Shot Detectors与OpenCV的dnn模块（检测图像中物体的模块）进行基于深度学习的物体检测，并将检测到的物体的名称作为一种信息发送回来。对用户的命令做出反应，在这种基于深度学习的方法中，有许多可以应用的物体检测方法。下面列出了三种常用的方法

a．Faster R-CNN.

b. You Only Look Once-YOLO.

c．Single Shot Detectors-SSD.

Faster R-CNN具有复杂的结构，难以实现和训练。更快的R-CNN (R for Region Proposal)算法的实施通过消耗更多的计算时间来产生结果。在我们的方案中，我们的资源是有限的，而智能音箱的用户将期望对命令有更快的反应。因此，R-CNN并不适合于智能音箱的应用。在另一端，YOLO v3[11]是一种最先进的物体检测方法，它的表现优于其他检测器[12]。在这里，智能音箱使用基于Open CV和深度学习的物体检测方法，结合SSD和移动网络来检测物体。这个方法可以检测到多达20个常见的日常物体和背景作为奖励·MobileNet SSD在Coco数据集(Common objects in Context)上进行训练，然后进行微调，以达到72.7%的mAP。可以检测到的物体的清单已经列出。

飞机﹑自行车﹑鸟﹑船﹑瓶子﹑公共汽车﹑汽车﹑猫、椅子、牛、餐桌、狗、马﹑摩托车﹑人﹑盆栽﹑羊﹑沙发﹑火车和电视显示器[12]。

## 2.6 亚马逊应用程序和服务开发者控制台的JSON脚本

一个JSON脚本被托管在Alexa开发者控制台。这个脚本通过使用HTTP over SSL/TLS的请求-响应机制与运行在Pi上的星期五定制技能脚本Alexa进行通信[13]。当注册的智能音箱用户使用语音命令与Alexa互动时，这个JSON脚本会理解用户的语音并从语音命令中提取独特的关键词。如果提取的关键词与预定义的关键词相匹配，那么JSON脚本就会向在Pi上运行的名为Alexa、"星期五"的自定义技能脚本发送一个带有JSON主体的POST请求(在第2.7节讨论）。图5说明了JSON脚本对这个智能音箱的作用。在这个用例中，Alexa|问“星期五”她“看到了什么”，这个命令需要被ISON脚本处理和理解，以便产生响应并将其发送到Pi上。

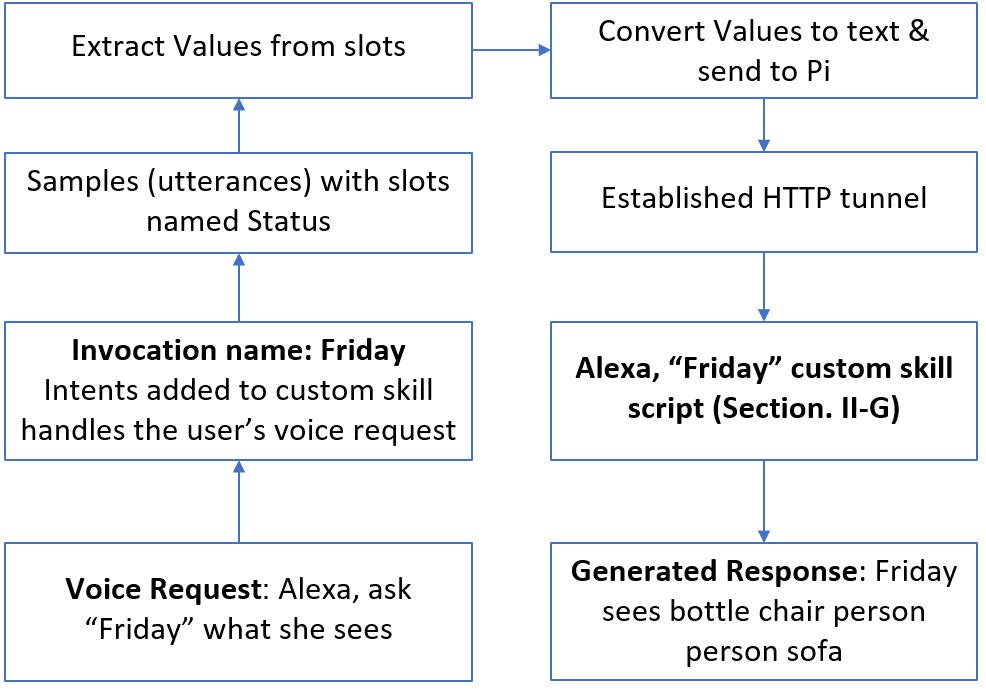


图 5 流程图说明了JSON定制技能脚本的运作

## 2.7 Alexa，"星期五"自定义技能脚本启动智能音箱上的物体检测

语音命令被用来激活智能音箱原型中加载的功能和自定义技能。这些语音命令是由用户通过Alexa Android应用程序或直接与智能音箱原型互动来给Alexa的。每个语音命令都有插槽，可以找到激活自定义技能所需的关键词。用户呼出的命令被发送到JSON脚本中，该脚本在2.6节中讨论。

该脚本从语音命令的插槽中提取独特的关键词，并通过建立的HTTP通道将提取的关键词发送到Raspberry Pi。在这种情况下，语音命令是Alexa，问"星期五"她"看到"什么。语音命令中的“看到”是将被JSON脚本拾取的关键词，并发送给Pi以激活物体检测脚本。当Pi收到这个关键词时，它就会检测其附近的物体，并以检测到的物体名称作为回应。在Pi上运行的星期五自定义技能python脚本收到这个关键词，并激活物体检测自定义技能。图6说明了接收语音输人﹑提取关键词并将关键词发送到Pi以激活自定义技能的过程。

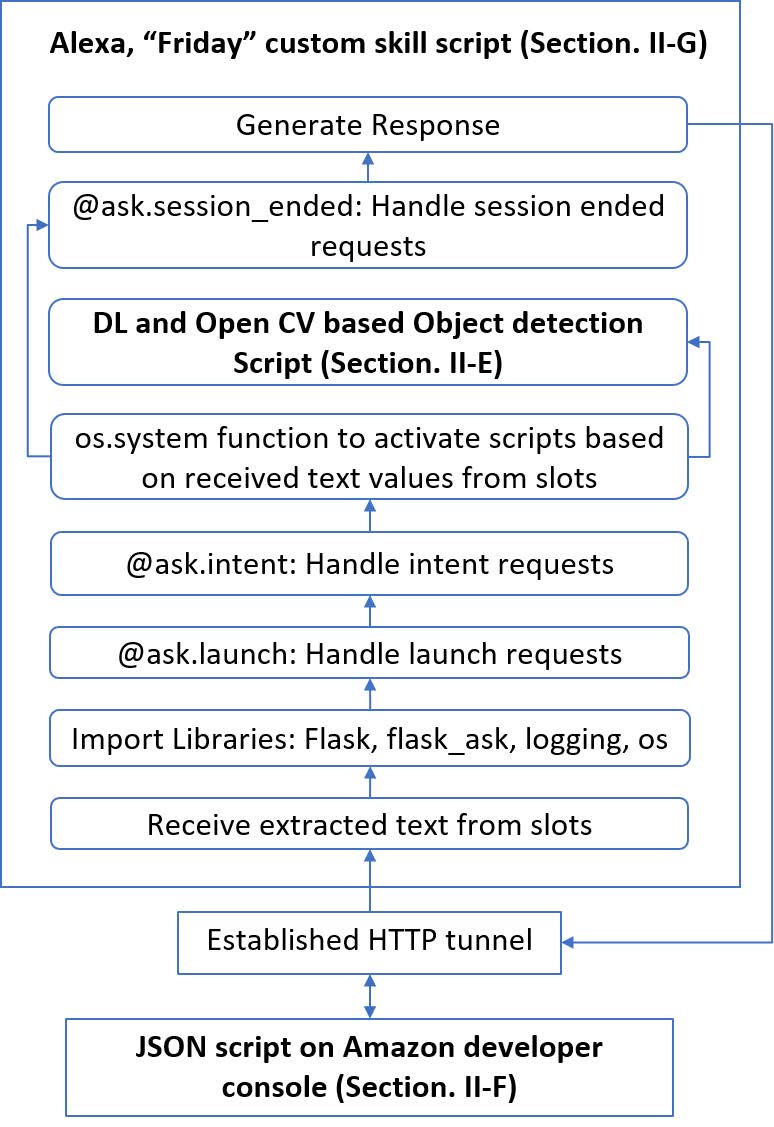


图 6 在Pi 上运行的Alexa，“星期五”自定义技能脚本的流程

# 结果和讨论

智能音箱的原型在作者家的起居室里进行了测试·由于用于物体检测的MobileNet SSD的平均精度 (MAP)已经知道是72.7%[12]，为了简洁起见，不讨论上述环境中的多个物体检测结果。由智能扬声器进行的物体检测原型图见图7。这里检测到的物体是一个瓶子﹑椅子﹑人、另一个人和沙发。如图8所示，在检测到的物体上画出了边界框，检测到的物体的类别标签及其相应的置信度分数被打印在图像上，同时也显示在Raspberry Pi的终端。边界框的坐标是使用单次检测器--SSD物体检测获得的，这一点在2.5节中讨论过。

最后，当用户喊出Alexa的命合“问星期五她看到了什么”时，来自第二节的脚本就会被激活。2.5、2.6和2.7节的脚本执行它们的任务，并向用户提供智能扬声器检测到的物体的名称。在这种情况下，Alexa回答说：“星期五看到了瓶子、椅子、人、沙发"。在测试过程中，我们注意到这里使用的相机模块没有对背光进行补偿（减少剪影)，以聚焦于前景图像。这影响了物体检测的性能，特别是在低光环境下。另外，摄像头模块需要大约460ms的预热时间（在理想的设置中)。然而，必须考虑到通信和处理的开销。因此，在捕捉图像进行进一步处理和检测之前，采用了480ms的相机预热时间（延迟）。

最近推出的Tensor Cam的AI摄像头﹑Toshiba’s Symbio[14]、Facebook’s Porta1[15]都是具有AI功能的摄像头智能音箱，这种具有摄像头的智能音箱将成为我们日常生活中不可或缺的一部分，并有望成为未来最热门的技术。这些支持摄像头的商用智能音箱对消费电子行业来说是新事物，它们仅将摄像头用于视频通话和媒体流。这些支持摄像头的商业智能音箱的未来版本肯定会拥有基于视频/图像的技能，如本工作中讨论的物体检测技能。

# 结论和未来发来方向

本文讨论了设计和开发一个最先进的支持摄像头的﹑基于麦克风阵列的Alexa智能音箱原型。除了基于麦克风阵列的Alexa互动之外，所设计的智能音箱系统还能够承载自定义技能，在用户要求时执行音频丛理﹑基于人工智能和计算机视觉的任务。一个这样的基于深度学习和计算机视觉的定制技能已经在智能音箱原型上实现和测试。

由于这个智能音箱的原型是一个基于摄像头的Linux系统。除了现有的物体检测自定义技能外，还可以开发和实现多种自定义技能。几个例子可以是，基于先进的图像必理（表情和手势检测）的Alexa的互动，根据识别的脸部来定制Alexa的回答等。我们还计划将我们现有的工作扩展到需要语音命令的多种用例，如智能企业（在线会议)[16]、智能制造(人机互动)[17]和数字语音助手。

**译文出处：**B. Sudharsan, S. P. Kumar and R. Dhakshinamurthy, "AI Vision: Smart speaker design and implementation with object detection custom skill and advanced voice interaction capability," 2019 11th International Conference on Advanced Computing (ICoAC), Chennai, India, 2019, pp. 97-102, doi: 10.1109/ICoAC48765.2019.247125.