

内克雷： 一种基于 Linux 内核的信息亭定制操作系统


孟加拉国美国国际大学的 A. S. M. 梅赫迪·哈桑·萨德

孟加拉国美国国际大学的 Md Mashrur Sakib Choyon

阿布·哈桑特·穆罕默德·里德万，孟加拉国美国国际大学

卡维什克·希克德，孟加拉国美国国际大学

乔杜里·阿克拉姆·侯赛因，孟加拉国美国国际大学

 <https://orcid.org/0000-0002-8769-2833>

摘要

近年来，自助服务终端设备在组织、机构或任何其他服务中心的信息传递方面的需求显著增加。它们已成为传统人工协助或接待台的更好替代品。然而，目前还没有适用于自助服务终端设备的动态操作系统或用户界面。本文介绍了一种名为“Nekray”的自助服务终端设备操作系统，该系统基于 Linux 内核环境开发。它旨在实现动态、快速、用户友好且具有交互性。开发的操作系统可根据需求更改其功能数据。它还支持即插即用功能，并且可以安装在任何低成本的硬件板上。此外，内置的人工智能也是该开发系统的一部分，通过图像处理来实现其功能。该系统在自助服务终端设备上维护用户数据的隐私和交互式传输。

关键词



3D 地图、3D 座位图、计算机视觉、动态系统、嵌入式多传感器、嵌入式系统、图像处理、即插即用、智能系统

介绍

智能化系统的应用在全球范围内正逐渐增多。这些设备的一些应用包括提供快速信息、自动化医疗设施以及家庭自动化。全球人口的不断增长也促使不同组织转向自动化系统以服务其用户。在这些设备中，自助服务终端因其易用性和无需任何外部帮助即可提供数据的能力而脱颖而出（Sad 等人，2020 年）。

自助服务亭是一种自动化电子设备，它利用各种现代硬件和软件来实现诸如导航、广告、接待和健康登记等广泛的功能（乔希等人，2017 年）。目前，它们几乎被应用于各个场所，从学校到大型办公室，以确保用户无需任何人工协助即可获得无缝的体验。据文献报道，自助服务亭已应用于市场（拉詹德兰，2018 年）、大学（埃克西奥格鲁等人，2018 年）、餐厅（韩等人，2019 年）以及医疗保健领域（阿夫扎利等人，2017 年；格拉迪亚和卡维亚，2017 年）。此外，研究人员还在评估自助服务亭在虚拟现实中的可用性。

DOI: 10.4018/IJERTCS.2021070104

 Copyright ©2021, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.


发展（格雷厄姆等人，2018 年）和实验室（布兰齐拉等人，2018 年）。另一种在不同组织中广泛使用以改进其运营的自助服务终端被称为信息自助服务终端。

操作系统是每个信息亭的重要组成部分，因为它们控制着系统的整体运行，而一个用户友好的操作系统能够轻松改善用户交互，这是任何信息亭的首要目标。对于信息亭而言，快速运行是必要的，因为大多数时候它们都放置在拥挤的区域。因此，选择合适的操作系统可能颇具挑战性。在这项研究中，开发了一种基于 Linux 内核的操作系统，能够应对信息亭中的各种挑战。该操作系统的硬件要求较低，几乎可以在任何系统上安装。本研究的主要贡献在于：

开发基于 Linux 内核的信息亭操作系统。

- 在操作系统中实现人工智能（AI）。

针对自助服务终端操作系统提升硬件资源损耗程度，以确保性能恒定。

-

开发一种能够根据用户需求变化的响应式用户界面。

所提出的解决方案主要关注用户在使用自助服务终端时的体验和舒适度。开发的操作系统消除了系统中不必要的后台进程，从而使设备在获取或传输数据时速度更快。系统的动态性使得可以根据任何组织的个性化需求来设计用户界面。此外，嵌入式系统与多种传感器的结合使设备具备更多功能。人工智能与吸引人的用户界面相结合，使该系统相较于市场上现有的自助服务终端设备更具独特性。该操作系统还通过减少设备的电力浪费来更高效地利用电力。诸如组织区域或建筑物的 3D 可视化等程序显著提高了用户对安装了 Nekray 操作系统的自助服务终端设备的满意度。由于对硬件配置的要求相当低，Nekray 操作系统能够在诸如树莓派这样价格更低且更易获取的微控制器上安装，从而具备了这一优势。因此，所提出的解决方案几乎可以在任何组织的自助服务终端设备中使用，安装成本更低，且在多个方面具有更大的优势。

本文的结构安排如下。“相关研究工作”部分概述了物联网领域的最新进展，并展示了 Linux 操作系统的不同应用。“操作系统架构”部分介绍了操作系统的开发情况。“自定义功能开发”部分展示了为信息亭开发的不同功能，使其能够在开发的操作系统上运行。“结果”部分展示了操作系统的各种操作和功能。最后，“结论”部分对整个研究进行了总结。

相关研究工作

随着物联网设备的不断增长，众多研究人员强调要为这些设备开发更友好、更高效的操作系统。因此，近来出现了前所未有的开发应用架构和框架的趋势。在（Kum 等人，2017 年）中，作者提出了一种物联网应用架构，该架构利用雾计算的概念，由物联网应用执行不同的服务。在另一项研究中，作者对物联网设备的各种操作系统进行了调查，以帮助该领域的研究人员为其应用选择合适的操作系统，该研究还总结了不同软件和硬件平台的特点（Afzal 等人，2019 年）。在（Choyon 等人，2020 年）中，使用树莓派和必要的传感器开发了一种基于物联网的健康监测设备。在（Payne 和 Abegaz，2018 年）中，分析了保护基于物联网的设备免受不同类型网络攻击的最佳实践。在另一项类似的研究中，对不同的最新操作系统进行了比较

在架构、内存、效率和实时性能方面已有所展示（Sabri 等人，2017 年）。还研究了逆向工程框架在基于 Linux 的操作系统中查找数据漏洞的潜力。该研究的作者成功地使用他们开发的框架对 WeMo 智能插头的通信协议进行了逆向工程，以分析其网络流量，从而检测并消除该系统中的任何设计缺陷（Liu 等人，2020 年）。

Linux 最近取得了巨大的飞跃，已成为广泛使用的操作系统之一，尤其是在物联网设备中（Serra 等人，2020 年）。在（Vishnubhatla，2020 年）中，展示了一个基于 Linux 服务器的票务亭，该票务亭使用处理器、基于 Qtopia 的图形用户界面和基于 MYSQL 的数据库服务器开发而成。在另一个应用中，基于 Linux 内核的虚拟机超调用被用于开发和实现一种虚拟机控制方案，该方案通过降低 CPU 负载显著提高了主机的效率（Choi 和 Hong，2019 年）。在（Boras 等人，2020 年）中，对三个 Linux 发行版进行了性能评估。为了测量性能，作者在相同的台式机上依次安装了这些发行版，结果表明 Pop!_OS 20.4 Linux 发行版表现最佳。基于 ARM 架构的设备大多使用不同 Linux 发行版的定制版本。在（Swain 等人，2020 年）中，作者还对物联网设备的嵌入式固件进行了比较分析。另一方面，张等人（2017 年）通过不同 Linux 内核版本的特性对其进行了描述，并提出了一种识别不同内核结构布局的方法。

操作系统架构

系统架构

现有的操作系统，如 Windows 和 Mac，由于其架构复杂，容易运行缓慢，这并不适合用于信息亭操作系统。为了解决这些问题，Nekray 基于 Linux 内核开发而成，其运行速度比其他操作系统快得多（Gerofi 等人，2018 年）。操作系统的开发取决于多个方面，包括硬件、内核和应用程序，如图 1 所示。在开发操作系统之前，需要确定硬件要求以确保其性能良好。另一方面，内核在任何系统中都充当操作系统与硬件之间的媒介（Høiland-Jørgensen 等人，2018 年）。在这种情况下，选择 Linux 内核是因为其性能、隐私保护以及开发新应用程序的灵活性（Passos 等人，2018 年）。

x64 架构

Nekray 的 x64 架构使其能够在能够同时处理 64 位进程的系统中运行。在为操作系统选择 64 位架构时，考虑了以下原因：

该操作系统是专门针对 64 位处理器开发的，这种处理器能够提供更大的地址空间，使用该操作系统，处理器能够寻址的内存容量远远超过 32 位处理器。这使得系统运行速度更快，整体性能大幅提升。

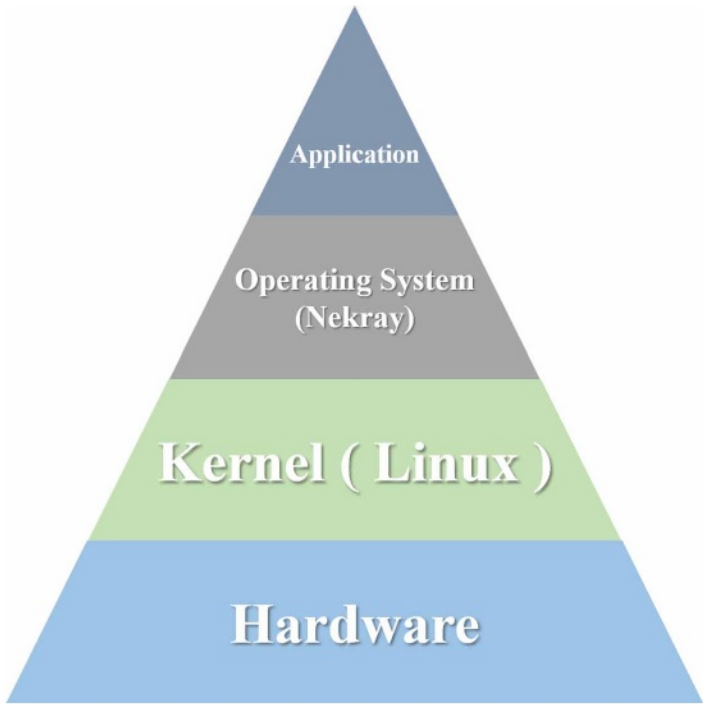
x64 架构使操作系统能够处理超过 4GB 的内存空间。因此，对于该操作系统的用户来说，内存空间没有限制。

使用内核补丁保护（KPP）能够降低有害软件对内核进行补丁修改的风险。

数据操作

安装完成后，Nekray 可以与自助服务终端的摄像头进行通信以做出决策。除了摄像头之外，还可以向自助服务终端添加其他传感器，操作系统能够轻松读取这些传感器的数据。目前，该操作系统无需任何额外编程即可支持以下传感器：

图 1.操作系统开发工作流程



- 指纹传感器
- 气压传感器
- 热传感器
- 气体传感器
- 湿度和雨量检测
- 加速度计模块
- 振动开关模块
- 近场通信（NFC）
- 湿度传感器
- 运动传感器
- 超声波传感器
- 射频识别卡读卡器
- 全球定位系统模块
- 陀螺仪
- 指南针
- 蓝牙/无线局域网/全球移动通信系统/红外二极管
- 心跳 / 脉搏传感器
- 光敏电阻器

硬件要求

Nekray 的开发方式确保了其安装更为简便，且运行该信息亭所需的硬件配置比大多数其他操作系统的要求要低得多（Sabri 等人，2017 年）。该操作系统的最低配置要求已在表 1 中列出。

表 1.Nekray 操作系统的硬件要求

硬件	最低要求
中央处理器	1 GHz
随机存取存储器	1GB
磁盘空间	35 吉字节

启动过程

图 2 展示了操作系统执行应用程序所遵循的步骤。首先，内核编译并检查必要的硬件。随后，运行必要的代码来编译操作系统。最后，应用程序可以从主屏幕启动执行。

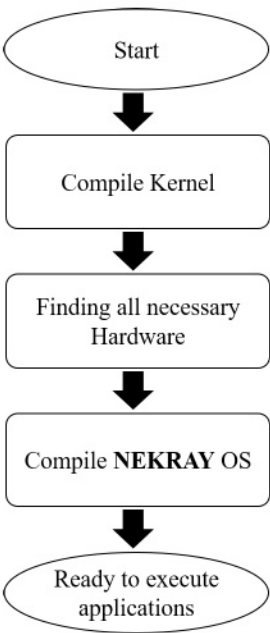
访问模式

Nekray 操作系统为用户和管理员提供了两种访问模式。这两种访问模式分别是“用户模式”和“开发者模式”。只有获得授权的人员才能访问“开发者模式”，而“用户模式”则可供所有人普遍使用。Nekray 操作系统的访问模式在保持动态性和用户友好性方面，几乎在市场上找不到其他类似的用于信息亭的操作系统能与之媲美。

用户模式

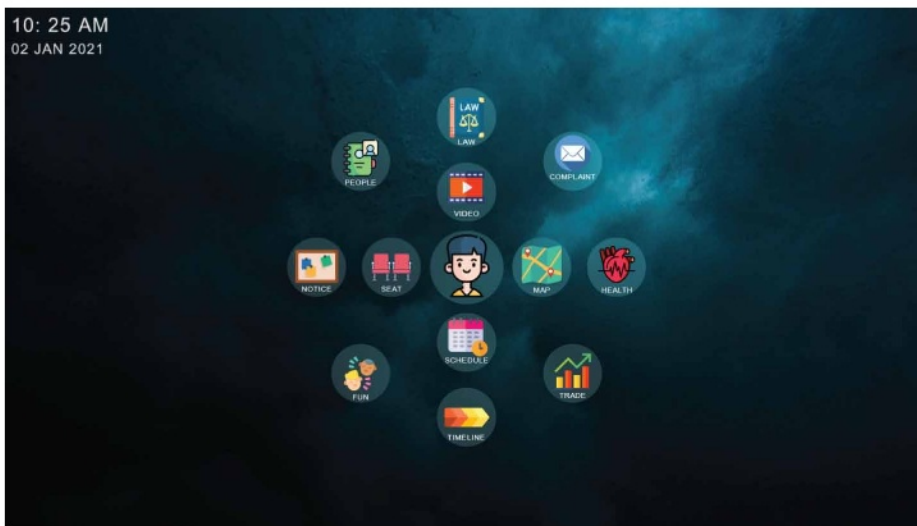
Nekray 操作系统的用户模式是为用户开发的系统主要界面，如图 3 所示。每当普通用户来到自助服务终端设备前，此模式就会被激活，并为用户执行不同的任务。在此模式下，用户无法对操作系统进行任何更改。根据用户类别（即访客用户、组织用户、授权用户）的不同，该模式会提供相应的服务。

图 2.应用程序执行



对于用户（Users），用户模式为其提供了丰富的功能。此外，根据用户的年龄和性别，主菜单中心的头像会相应地改变和匹配。这一检测由系统的图像处理功能完成。对于已授权用户，头像将被系统数据库中预先定义的用户个人资料图片所替代。另一方面，对于访客用户或数据库中未预先定义用户数据的用户，仅会显示头像。Nekray 操作系统为用户提供了极具吸引力的用户界面，这证明该系统非常用户友好、用户交互性强且易于使用，无需任何先验知识。

图 3. 用户模式下的主菜单



开发者模式

Nekray 操作系统为信息亭设备提供了现代时期一些最强大的功能。Nekray 操作系统的开发者模式专为组织的管理员或行政人员打造，让他们能够根据自身需求修改操作系统，并且可以自由更改系统的某些功能。该开发者模式主要基于 Linux 内核环境，与市面上现有的 Linux 发行版类似。因此，管理员可以根据需要轻松更改某些功能。管理员可以通过特定的快捷键访问开发者模式。对于 Nekray 操作系统，默认的快捷键设置为“Ctrl+Alt+S”。按下此指定的快捷键后，屏幕上会出现用户名和密码的面板，管理员需要输入预先指定的用户名和密码才能进入开发者模式。由于信息亭设备大多用于触摸屏显示器，普通用户无法通过触摸屏键盘访问开发者模式。需要手动将键盘插入信息亭处理器单元的硬件设备中，以便按特定的快捷键更改访问模式，从用户模式进入开发者模式。开发者模式使管理员能够访问 Nekray 操作系统的后端结构，并允许他们对系统进行任何必要的更改。通过开发者模式，管理员还可以对诸如 3D 地图、3D 座位图、活动、用户界面等功能进行更改。

定制功能开发

开发的 Nekray 操作系统能够根据任何信息亭的需求运行不同的应用程序。在这种情况下，为信息亭开发了不同的程序，这些程序可以在该操作系统架构上运行。这些程序/功能是使用多个框架（例如 OpenCV、Electron、MySQL 等）和编程语言（例如 Python、JavaScript、C# 等）开发的。尽管在系统开发中使用了多种框架和编程语言，但它们都被完美融合，以呈现最终产品。程序的用户界面和展示方式经过精心设计，使设备更具用户交互性和易用性。

在初级阶段，主要使用的硬件处理单元是树莓派（Freeborn, 2019）。然而，由于 Nekray 是基于 Linux 内核开发的，该系统可以安装在任何其他型号的处理单元或计算机硬件上（例如 Nvidia Jetson、华硕 Tinker Board S、Odroid、Libre Computer Board、Orange Pi、LattePanda 等）。要在计算机板（树莓派 4B）上运行不同类型的框架和编程语言，首先需要安装不同的框架和软件来搭建环境。这些框架包括 Electron、TensorFlow Lite、Darkflow、OpenCV、JavaScript 等。

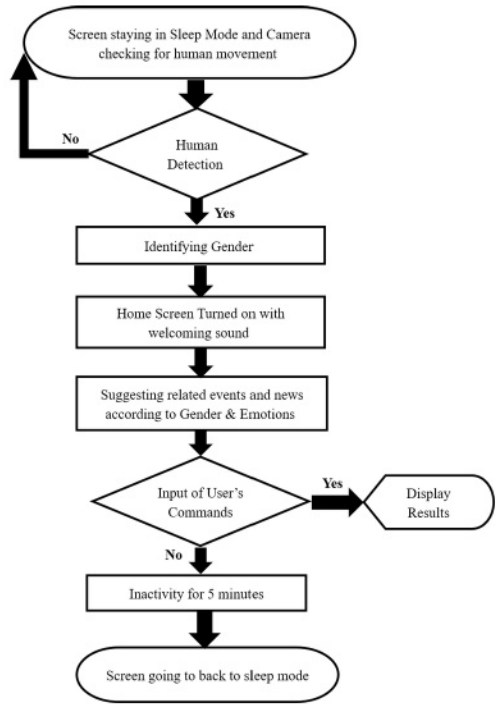
图像处理

Nekray 能够高效快速地执行图像处理功能。这些功能是使用 Python 框架中的 OpenCV、Tensorflow、YOLOv2、COCO 数据集等开发的。Nekray 在设备后端执行此图像处理功能，分为三个部分：1) 人体检测，2) 面部识别，3) 年龄和性别预测。这些部分是使用单独的条件程序和库开发的。在图像处理的这一部分，Nekray 操作系统是基于 Python 框架中的 OpenCV 开发的。它能够实现实时计算机视觉功能，且精度极高。基本上，在这种情况下，OpenCV 利用 SSD 框架以及 ResNet 作为核心网络。在我们的项目中，该库使用了先前收集的人体数据集。该系统在检测人体方面被证明是高效的。人体检测算法的开发方式是，当相机范围内有人体存在时，只要检测水平达到至少 60%，系统就会通过图像处理识别出人体。考虑到摄像头性能在低光照条件下常有波动，检测算法的要求被适当降低以确保一致性。检测完成后，系统会识别出被检测用户的性别和情绪。该系统的隐藏功能之一是通过机器学习，根据用户的性别和当前情绪状态（由用户面部图像确定）生成新闻和事件通知。这种图像处理功能使系统既节能又具有用户交互性。由于设备的主要耗电部分是显示屏本身，所以当附近无人时，系统会处于睡眠模式。一旦检测到有人在设备的触摸距离内，显示屏就会亮起，并伴有欢迎声向用户致意。如果用户向设备发出获取信息的指令，设备会执行该操作，并在屏幕上显示所需结果。当用户停止使用该设备后，设备将等待 5 分钟以检测用户发出的进一步指令或检测附近是否有其他人的存在。如果未检测到人的存在，屏幕将再次进入休眠模式，并持续扫描是否有人出现。整个流程的流程图已在图 4 中展示。这一开发有助于设备减少能耗，同时避免了设备的电力浪费。

人体检测

该操作系统在开发时就已设定，安装完成后，系统能够检测到前方有人，随即点亮显示屏并发出问候声。在此过程中，

图 4.图像处理流程图



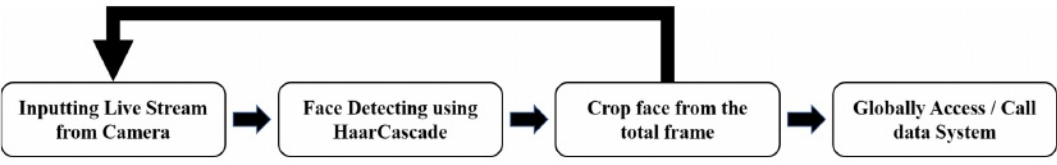
基本的图像处理程序被编写用于在一定距离外检测到。为此，采用了目标检测方法检测人。OpenCV 和 COCO 数据集相结合，有一个默认的数据集，通过图像处理能够识别多达 93 种不同的物体。利用这个数据集，系统执行人体检测。此方法如图 5 所示。

人脸识别

在构建了图像识别的初始环境之后，Nekray 接着被训练用于人脸识别。在某些情况下，Nekray 会使用大量样本进行训练，以识别特定的人脸并确定其姓名和其他信息。为此功能，采用了计算机视觉架构来识别人脸。开发此程序时使用了一些高效的库，其中以下库可被突出视为最常用的库：

- 来自 GitHub 的 “face_recognition” 库
- “Dlib C++ 库”
- 《scikit-learn：Python 中的机器学习》

图 5. 人体检测工作流程



在实时检测中，Nekray 利用计算得出的 128 维人脸嵌入向量从图像中检测人脸 (Rosebrock, 2020)。这些图像由系统从实时视频数据中采集，然后从每一帧中分离出来。此外，Nekray 还通过 128 维嵌入向量用支持向量机（也称为“SVM”）进行训练。这样，该系统就能对用于量化特定图像中每张人脸的 128 维特征向量进行提取，其中借助深度学习，采用了 Python 和 Torch 的组合实现 (Schroff 等人, 2015)。通过这种模型，Nekray 在仅使用 128 维的情况下，在表示和执行最先进的面部识别方面达到了更高的效率。

在实际操作中，图像识别和人脸检测遵循了几个步骤。首先，需要向 Nekray 提供一张图像或一段视频作为输入以启动处理过程。一旦输入被提供给面部识别流程，Nekray 就开始对其进行分析，然后应用面部检测过程，以确定该特定图像中人脸的位置。对于视频，系统会逐帧将其拆分为图像。为了提高系统的效率，Nekray 可以选择计算“面部特征点”，这使其能够对人脸进行预处理和对齐 (Johnston & de Chazal, 2018)。这种特定的人脸对齐过程在某些流程中提高了面部识别的准确性。在应用人脸对齐（这只是一个可选步骤）以及裁剪之后，作为输入的人脸图像将通过 Nekray 的深度神经网络。借助 FaceNet，Nekray 可以直接从人脸图像学习到紧凑的欧几里得空间映射。在这里，坐标之间的距离直接对应于人脸相似度的度量。

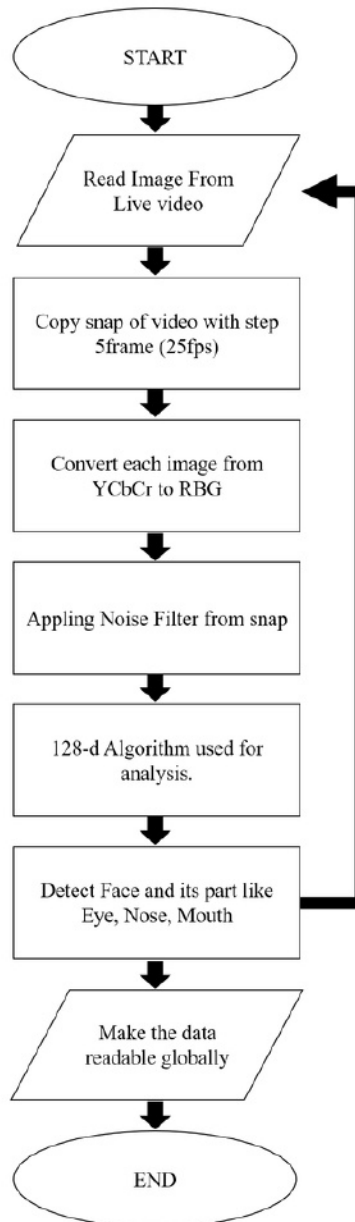
按照检测人脸并提高识别准确率的要求设置好程序后，Nekray 进一步开发，使其能够通过特定数据集进行训练，并通过识别人脸来识别某些用户。为了使用深度学习训练人脸识别模型，每个输入数据批次都像处理三种类型的图像那样进行处理。在此阶段，系统为每张人脸提取 128 维嵌入。之后，需要使用机器学习的“标准”模型（如支持向量机、K 近邻、随机森林等）在嵌入层上对 Nekray 进行训练。我们使用了一些数据集和人类受试者来训练该系统。整体工作流程已在图 6 中展示。

用于训练 Nekray 系统的三种类型的图像可以用以下几点来定义：

- **锚点：**它是系统中具有身份标识的当前人脸的边缘或锚点。Nekray 系统会在人脸周围生成一个具有特定高度和宽度的锚框。然后利用此框来定义特定人脸图像的纵横比、缩放比例以及其他几何属性。接着，它会在系统中与该人脸数据库相关联的身份名称中识别出该对象。
- **正面图像：**正面图像聚焦于所识别的图像（例如：面部），并调整图像的一些属性（例如：对比度、饱和度、锐度、梯度），然后将主体图像的聚焦区域变暗以识别面部。换句话说，它通过将聚焦区域转换为变暗的黑色区域，然后根据数据集识别面部。
- **负面图像：**负面图像的工作原理与正面图像类似。它同样聚焦于给定图像中识别出的图像（例如“面部”），并调整一些图像属性（例如对比度、饱和度、锐度、渐变），然后提亮聚焦区域。换句话说，它通过将聚焦区域转换为亮白色区域来处理该区域，然后根据数据集识别面部。

《国际嵌入式与实时通信系统杂志》
第 12 卷 · 第 3 期 · 2021 年 7 月 - 9 月

图 6.人脸识别工作流程



Nekray 在图像处理方面的整体准确性取决于其训练数据量。在有足够数据的情况下，Nekray 系统运行效率更高，在检测用户方面表现更佳。

年龄检测

Nekray 图像处理新增的一项高级功能是“年龄检测”。Nekray 系统能够从图像或实时视频流中识别并预测出一个人的年龄，如图所示。

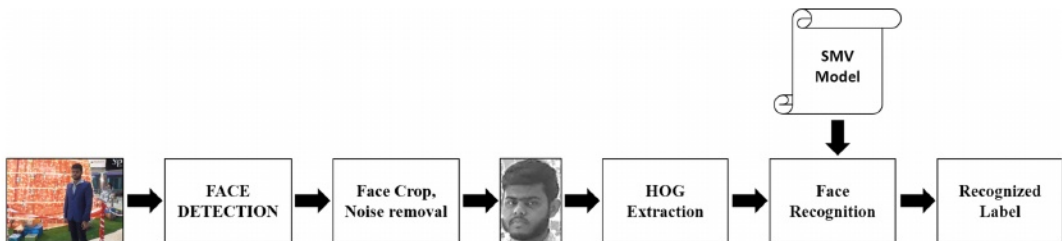
7. 这是一个仅通过分析人脸照片就能自动检测出一个人年龄的过程。此功能是通过计算机视觉技术实现的。

年龄检测功能分两阶段开发：

阶段 1：在年龄检测过程的第一阶段，系统从作为输入提供给系统的图像/视频流中检测人脸。

第二阶段：在年龄检测流程的第二阶段，系统提取面部感兴趣区域（ROI），并应用年龄检测算法来确定人员的年龄。

图 7. 年龄检测



在第一阶段执行任务时，需要任何能够为照片中的人脸生成边界框的人脸检测器。有许多人脸检测算法能够完成此操作，例如 Haar 级联、HOG + 线性 SVM、单次检测器（SSD）等。

Haar 级联算法的运行速度要快得多，能够在嵌入式设备上实时运行（Mantoro 和 Ayu, 2018）。但 Haar 级联算法的缺点是检测结果不够准确，而且出现误报的概率较高。HOG + 线性 SVM 模型的检测准确率高于 Haar 级联算法，但处理速度相对较慢。此外，它们对遮挡的容忍度较低，可能会产生面部不完整的检测结果。而且由于视角变化（即从不同角度拍摄面部图像会产生不同的视图），该模型也无法产生准确的结果。

基于深度学习开发的人脸检测器比其他检测器更稳健，也更有可能实现更高的准确率。但与 Haar 级联和 HOG + 线性支持向量机相比，它们需要更多的计算资源。综合考虑这些因素，并根据人脸检测器的速度和准确率进行判断，Nekray 系统采用了 SSD（单次检测）模型。这是一种基于深度学习的模型，需要大量的计算资源，但在人脸检测方面提供了显著更高的准确率。该模型通过逐帧处理视频流来分割图像，因此在每一帧中，它接收一张单独的图像，然后运行算法过程来检测人脸。这种方法在人脸检测方面被证明非常有效，最终有助于第二阶段对人的年龄进行检测。

一旦人脸检测器生成了图像/视频流中人脸的边界框坐标，系统就可以进入处理流程的第二阶段，即对人的年龄进行识别。从给定的图像边界框中，首先利用生成的人脸框的（X, Y）坐标提取人脸感兴趣区域（ROI）。在此过程中，算法会忽略图像/帧的其余部分。通过此过程，系统使年龄检测器能够专门聚焦于人的面部，而非图像中的其他任何区域。

在进行面部检测之后，将不会对图像中的任何其他无关噪声进行处理。然后，将面部区域（ROI）传递给模型，对人员进行真实预测。目前有许多年龄检测算法可用，但基于深度学习的年龄检测器比其他任何算法都更受欢迎且更高效。对于 Nekray 系统，选择了一个基于深度学习的年龄检测模型[8]。这里使用了一个类似于 AlexNet 的简单架构，经过训练过程来学习并检测总共 8 个年龄区间。这些年龄区间分别是：（1）0 - 2 岁，（2）4 - 6 岁，（3）8 - 12 岁，（4）15 - 20 岁，（5）25 - 32 岁，（6）38 - 43 岁，（7）48 - 53 岁，以及（8）60 - 100 岁。这些年龄区间被证明是不连续的，这是有意为之，因为用于训练模型的“Adience 数据集”就是如此定义年龄范围的。对于 Nekray 系统，使用了一个预训练的年龄检测模型。

嵌入式多传感器

为了收集和提供实时的环境及生物数据，开发了 Nekray 系统，该系统能够同时控制多个传感器，并有效处理模拟信号和数字信号，如图 8 所示。为了控制这些多个传感器，为该系统设计了一个信号转换器，将模拟信号转换为数字信号，然后输入到系统 CPU 进行进一步处理。为 Nekray 操作系统开发的这种转换器被命名为“**Nekray 模块**”。该 Nekray 模块可以替换为任何其他微控制器（例如 Arduino Uno、Arduino Mega 等），但需要根据相应的必要平台（例如 Arduino CC）对微控制器进行编程。Nekray 模块连接到多个传感器，这些传感器收集不同类型的生物和环境数据。收集到的数据将以模拟信号或数字信号的形式传输到 Nekray 模块。该模块会将所有接收到的信号转换为数字信号，并通过串行通信将其发送到系统 CPU。对于 Nekray 操作系统，为与信息亭设备一起使用的指定传感器开发了相关代码。为各个传感器指定的不同代码被合并为一个统一的代码，能够同时控制所有传感器。Nekray 操作系统可使用的传感器数量并非固定不变。相反，这一特性也保持了动态性，即在需要时，可在任何给定时间将所需的传感器添加到系统中，并对代码和电路图进行必要的修改。

3D 地图开发

3D 地图开发是引入 Nekray 系统的最独特功能之一。这使用户能够通过地图导航，了解所需目的地的位置并

图 8. 嵌入式多传感器的工作流程

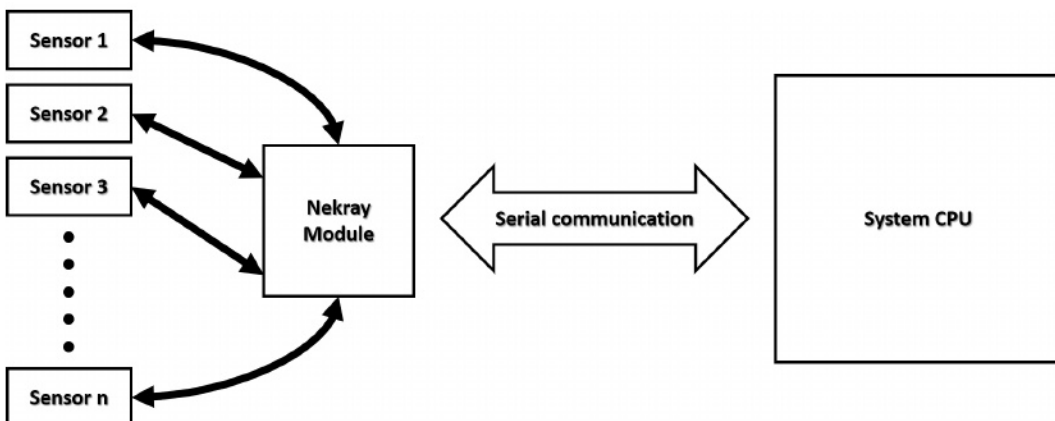
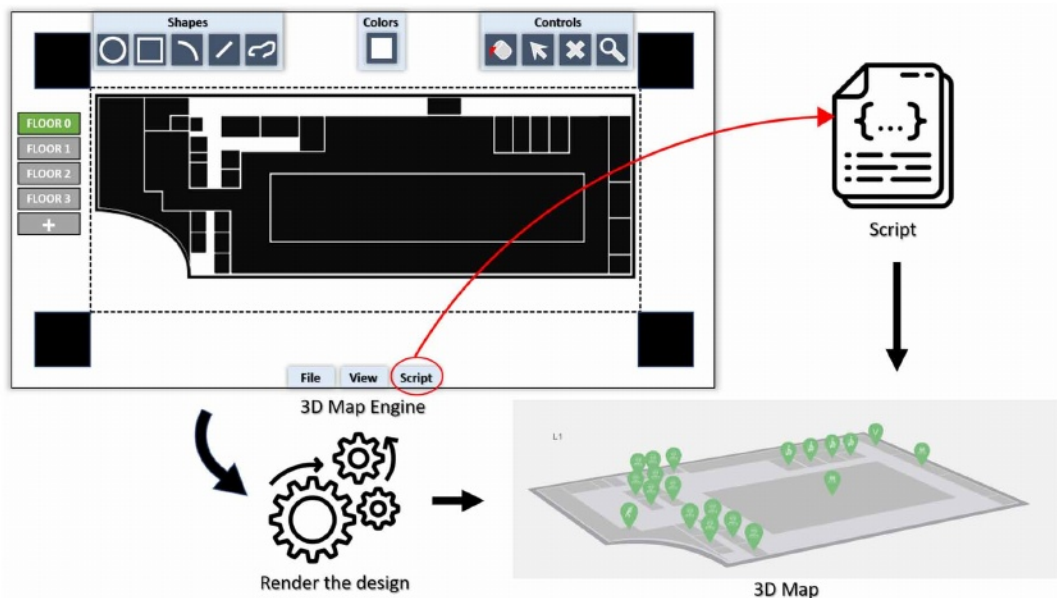


图 9.Nekray 中 3D 地图的开发



如何到达那里。该 3D 地图主要使用 JavaScript 编程开发而成。3D 地图的开发分为三个阶段。以下各节将对这些阶段进行说明。

结构建筑设计

在现阶段，模型组织构建的整个架构是使用 Nekray 开发者模式中默认提供的 3D 地图引擎设计的。结构设计是根据模型建筑各楼层和空间的实际尺寸创建的。最终开发出的建筑设计随后会通过 3D 地图脚本，利用各楼层的尺寸来创建 3D 地图。

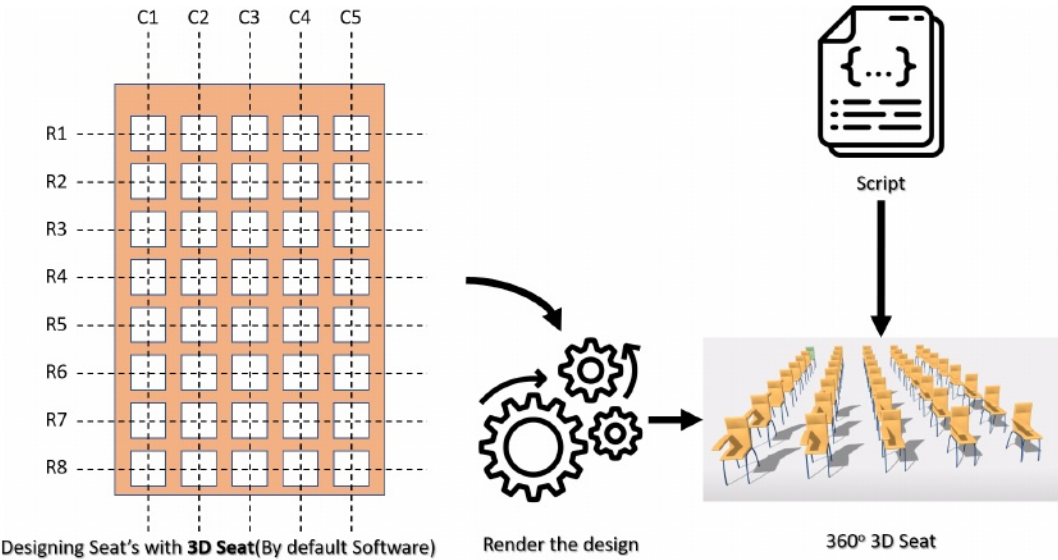
三维地图数据库

在这一特定阶段，模型构建所需的数据被精确收集。这些数据包括以下内容：房间号、人员姓名、房间名称、人员的电子邮件地址、人员在组织中的职位等。这些数据是根据目标组织或机构的数据提供的，并用于开发 3D 地图的数据库。之后，这些数据将用于开发包含所有这些数据的 3D 地图的最终版本。用户在使用 3D 地图功能时，可以获取地图上任何位置的信息，还能了解如何到达那里。

编程与开发三维地图

这是该功能的最后阶段，前两个阶段的所有数据在此汇聚并用于开发最终程序。基本上，前两个阶段获取的所有数据都被用于程序的编码。这些编码最终生成了最终的 3D 地图。用户可以根据自身需求对 3D 地图进行导航。Nekray 提供的一项独特功能是 3D 地图的动态性。根据任何组织的数据，3D 地图可以随时轻松地进行设计和修改。3D 地图开发的整体流程已在图 9 中展示。

图 10. 在 Nekray 中 3D 座位布局的发展



三维座位图的开发

3D 座位规划功能也是基于与图 10 所示的 3D 地图相同的工作或概念开发的。在这个程序中，设计了一个具有给定座位数量的默认房间模型。在这种情况下，房间和座位的结构也是动态的，可以根据任何组织或机构（大学、机场、汽车站、火车站、办公室、会议厅等）进行更改。在这个功能中，首先通过 2D 图形使用 3D 座位引擎设计座位。然后，复制单个座位或椅子的设计，以在房间中生成任意给定数量的座位。在程序中开发了一个位置矩阵，用于识别教室中每把椅子的位置。还编写了一个后端数据库，其中包含所有用户的数据。每当在系统中搜索任何特定用户的数据时，程序都会获取该用户的数据，并在房间中定位该用户分配的座位。

结果分析

本节通过计算所用算法的准确率，并将所开发的操作系统与现有操作系统进行性能对比，来评估所开发操作系统的性能。为了测试操作系统所使用的人工智能的准确性，我们测量了准确率、性能、召回率和 F1 分数。此外，为了证明开发适用于信息亭的新操作系统的合理性，我们重点关注了现有操作系统的资源利用率，并将其与 Nekray 操作系统进行了比较。

表 2.人体检测数据集

任务：人体检测	实际班级	预测类别		
			类别 = 是	班级 = 否
		类别 = 是	真阳性 (TP) = 47	假阴性 (TN) = 8
		班级 = 否	假阳性 (FP) = 4	真阴性 (TN) = 63

表 3.性别检测数据集

任务: 性别检测	实际班级	预测类别		
			类别 = 是	班级 = 否
		类别 = 是	真阳性 (TP) = 36	假阴性 (FN) = 1
		班级 = 否	假阳性 (FP) = 16	真阴性 (TN) = 59

AI性能

人工智能是 Nekray 操作系统的核心功能之一，用于检测人类并以更简便、更友好的方式向他们提供所需信息。如前文所述，Nekray 通过深度学习借助支持向量机（SVM）对人脸进行识别。为了测试其性能，总共使用了 122 个数据集，如表 2 所示。这些数据集用于测试算法正确预测实际对象的能力。

利用表 2 中的数据集，可以得到以下结果：

准确率 = 真阳性 + 真阴性 / 真阳性 + 假阳性 + 假阴性 + 真阴性 = 0.9016
精确率 = 真阳性 / 真阳性 + 假阳性 = 0.9215

召回率 = 真阳性 / （真阳性 + 假阴性）= 0.8545

F1 分数 = 2 × （召回率 × 准确率） / （召回率 + 准确率）= 0.8867

这些结果表明，该操作系统在大多数情况下能够准确地完成人体检测任务。

性别预测

Nekray OS 利用摄像头数据来检测用户的性别和情绪，以便系统能够对用户进行画像。这样，系统就能向用户推荐相关的新闻和事件。在这一阶段，将对这项任务的准确性进行评估。与人体检测采用相同的策略，使用了类似的测试数据集来评估算法预测对象性别的能力，如表 3 所示。

从检索到的数据中可以得出以下结果：

准确率 = 真阳性 + 真阴性 / 真阳性 + 假阳性 + 假阴性 + 真阴性 = 0.7786
精确率 = 真阳性 / 真阳性 + 假阳性 = 0.6923

召回率 = 真阳性 / （真阳性 + 假阴性）= 0.7659

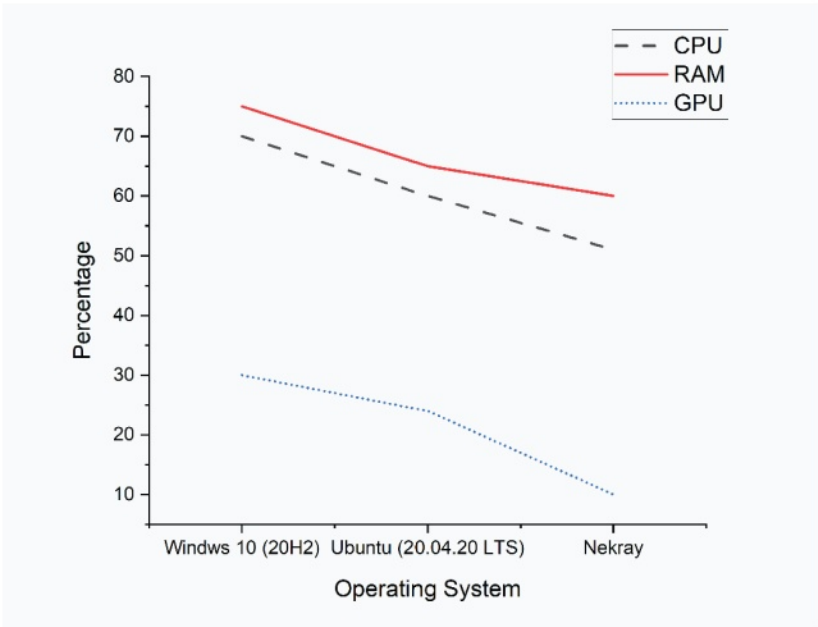
F1 分数 = 2 × （召回率 × 准确率） / （召回率 + 准确率）= 0.7272

根据所获得的结果，可以说 Nekray OS 的性别检测准确率相当高，尽管其准确率低于人体检测。

硬件资源利用率

图 11 展示了信息亭运行期间，使用三种不同操作系统（Windows 10、Linux、Nekray）的不同硬件的资源利用率。这些数据是在信息亭运行于由 1.5GHz 四核 ARM Cortex A72 芯片组、4GB 内存和 164MB 显存组成的硬件上获取的。在早期的研究（Boras 等人，2020 年）中也进行了类似的评估。为了获得准确的数据，每个操作系统都单独执行了相同的任务。结果表明，在信息亭运行时，Nekray 操作系统使用的 CPU、RAM 和 GPU 内存都比其他操作系统少。这对于信息亭来说非常有用，因为信息亭是为持续运行而设计的，不能有任何中断。

图 11. 不同操作系统下的硬件利用率



调查结果

为了解用户对开发的 Nekray 操作系统的体验，开展了一项调查。超过 1500 人参与了此次调查，他们均在使用搭载 Nekray 操作系统的开发款自助服务终端后给出了实时反馈。参与者主要是某所大学（孟加拉国美国国际大学）的学生和来访的客人。如图 12 所示，绝大多数人（91%）认为该自助服务终端比现有的更易于使用。这是

图 12. 关于 Nekray 操作系统易用性的调查结果

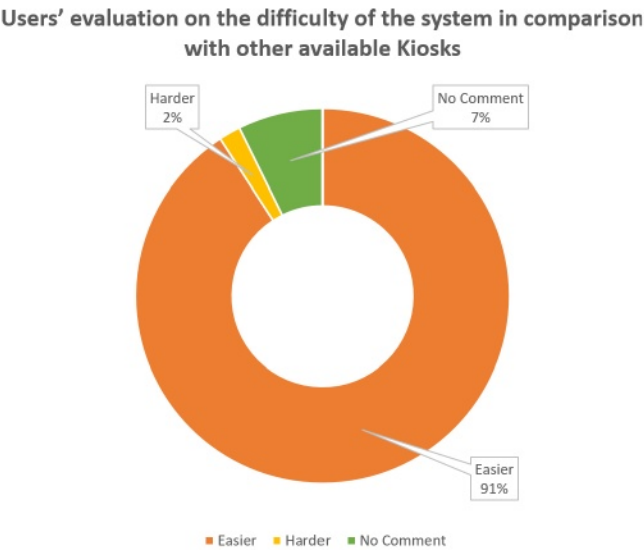
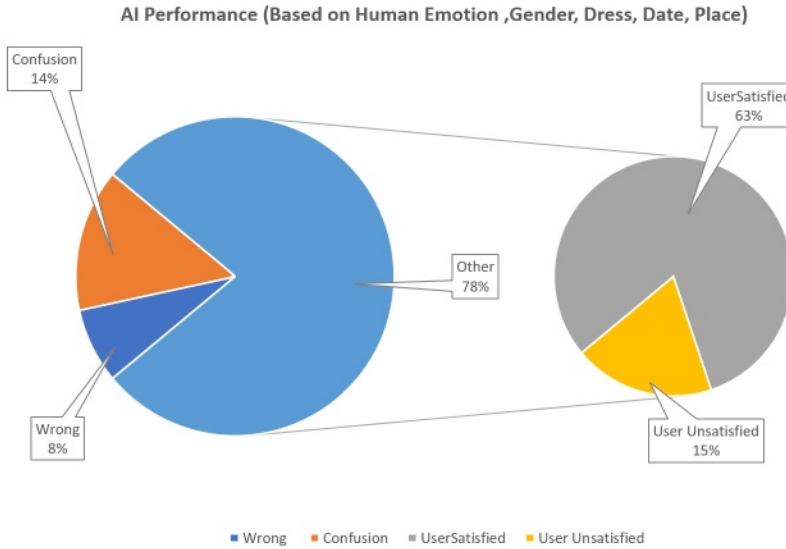


图 13. 关于 Nekray OS 人工智能性能的调查结果



开发该操作系统时的主要目标是打造一个用户友好型操作系统。如图 13 所示，63% 的用户对自助服务终端内置的人工智能功能检测其面部和情绪的准确性感到满意。

结论

本文介绍了一种基于 Linux 内核的用于信息亭设备的操作系统的开发情况。在对现有操作系统进行调研之后，对所开发操作系统的各个方面进行了广泛描述。还根据信息亭的需求开发了可在该操作系统上运行的不同功能。由于信息亭需要为用户提供快速准确的数据，因此该操作系统基于 Linux 内核开发，因其性能和易用性而广受欢迎。开发此操作系统的关键动机在于减少信息亭运行期间的硬件资源使用，降低此类信息亭设备的安装成本，并打造一个吸引人的用户界面以提升用户体验。该目标已成功实现，因为该操作系统在执行日常操作时比现有操作系统使用的硬件资源显著减少。该操作系统还成功地利用诸如支持向量机（SVM）等不同算法识别了人体动作和性别，准确率分别达到了 90% 和 78%。该开发的操作系统在用户界面、硬件利用以及机器学习应用方面与现有的信息亭操作系统有所不同。这项研究的未来前景广阔，操作系统的图像处理能力可以得到提升，未来还可以研究人工智能对降低电子设备能耗的影响。

参考文献

- Afzal, B., Umair, M., Asadullah Shah, G., & Ahmed, E. (2019). Enabling IoT platforms for social IoT applications: Vision, feature mapping, and challenges. *Future Generation Computer Systems*, 92, 718–731. doi:10.1016/j.future.2017.12.002
- Afzali, M., Ahmadi, M., & Mahmoudvand, Z. (2017). Data Requirements and the Basis for Designing Health Information Kiosks. *Acta Informatica Medica*, 25(3), 198. PMID:29114115
- Boras, M., Balen, J., & Vdovjak, K. (2020). Performance Evaluation of Linux Operating Systems. *2020 International Conference on Smart Systems and Technologies (SST)*, 115–120. doi:<ALIGNMENT.qj></ALIGNMENT>10.1109/SST49455.2020.9264055
- Branzila, M., Sarmasanu, C., Ciudin, D., & Lacatusu, D. (2018). Sensors and transducers laboratory kiosk. *2018 International Conference and Exposition on Electrical And Power Engineering (EPE)*, 377–380.
- Choi, E., & Hong, J. (2019). Design and implementation of virtual machine control and streaming scheme using Linux kernel-based virtual machine hypercall for virtual mobile infrastructure. *Proceedings of the Conference on Research in Adaptive and Convergent Systems*, 57–60. doi:<ALIGNMENT.qj></ALIGNMENT>10.1145/3338840.3355690
- Choyon, M. M. S., Rahman, M., Kabir, M. M., & Mridha, M. F. (2020). IoT based Health Monitoring Automated Predictive System to Confront COVID-19. *2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*, 189–193. https://doi.org/ doi:10.1109/HONET50430.2020.9322811
- Ekioğlu, M., Güler, H., Terzi, F., Yıldırım, H. S., & Yücel, B. (2018). *UXD for a Prototype Campus Information Kiosk*. Academic Press.
- Freeborn, T. J. (2019). Performance evaluation of raspberry Pi platform for bioimpedance analysis using least squares optimization. *Personal and Ubiquitous Computing*, 23(2), 279–285.
- Gerofi, B., Riesen, R., Takagi, M., Boku, T., Nakajima, K., Ishikawa, Y., & Wisniewski, R. W. (2018). Performance and scalability of lightweight multi-kernel based operating systems. *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 116–125.
- Gladia, R. A., & Kavya, G. (2017). Design and Development of Non-Invasive Kiosk for Self-Care Health Management. *Global Journal of Pure and Applied Mathematics*, 13, 4787–4794.
- Graham, K., Fai, S., Dhanda, A., Smith, L., Tousant, K., Wang, E., & Weigert, A. (2018). The VR kiosk. In *Digital Cultural Heritage* (pp. 324–336). Springer.
- Han, J., Oh, Y., & Ham, S. (2019). Influence of Ordering Kiosk Nutrition Information Transparency and Information Quality on the Customer Behavioral Intention in Fast Food Restaurants. *Journal of the Korean Dietetic Association*, 25(3), 165–177.
- Høiland-Jørgensen, T., Brouer, J. D., Borkmann, D., Fastabend, J., Herbert, T., Ahern, D., & Miller, D. (2018). The express data path: Fast programmable packet processing in the operating system kernel. *Proceedings of the 14th International Conference on Emerging Networking EXperiments and Technologies*, 54–66.
- Johnston, B., & de Chazal, P. (2018). A review of image-based automatic facial landmark identification techniques. *EURASIP Journal on Image and Video Processing*, 2018(1), 86.
- Joshi, A., Arora, M., & Malhotra, B. (2017). Usability Evaluation of a Portable Health Information Kiosk Using a SMAARTTM Intervention Framework. *Global Journal of Health Science*, 9(8), 153. doi:10.5539/gjhs.v9n8p153
- Kum, S. W., Moon, J., & Lim, T. (2017). Design of fog computing based IoT application architecture. *2017 IEEE 7th International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*, 88–89. doi:<ALIGNMENT.qj></ALIGNMENT>10.1109/ICCE-Berlin.2017.8210598
- Liu, K., Yang, M., Ling, Z., Yan, H., Zhang, Y., Fu, X., & Zhao, W. (2020). On Manually Reverse Engineering Communication Protocols of Linux Based IoT Systems. *IEEE Internet of Things Journal*, 1–1. doi:10.1109/JIOT.2020.3036232

- Passos, L., Queiroz, R., Mukelabai, M., Berger, T., Apel, S., Czarnecki, K., & Padilla, J. (2018). A study of feature scattering in the linux kernel. *IEEE Transactions on Software Engineering*.
- Payne, B. R., & Abegaz, T. T. (2018). Securing the Internet of Things: Best Practices for Deploying IoT Devices. In K. Daimi (Ed.), *Computer and Network Security Essentials* (pp. 493–506). Springer International Publishing. doi:10.1007/978-3-319-58424-9_28
- Rajendran, P. S. (2018). Virtual information kiosk using augmented reality for easy shopping. *International Journal of Pure and Applied Mathematics*, 118(20), 985–994.
- Rosebrock, A. (2020). *Raspberry Pi and Movidius NCS Face Recognition—PyImageSearch*. Pyimagesearch. <https://www.pyimagesearch.com/2020/01/06/raspberry-pi-and-movidius-ncs-face-recognition/>
- Sabri, C., Kriaa, L., & Azzouz, S. L. (2017). Comparison of IoT Constrained Devices Operating Systems: A Survey. *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, 369–375. doi:<ALIGNMENT.qj></ALIGNMENT>10.1109/AICCSA.2017.187
- Sad, A. M. H., Choyon, M. M. S., Rhydwani, A. H. M., & Hossain, C. A. (2020). An Interactive Low-Cost Smart Assistant System: Information Kiosk as Plug Play Device. *2020 27th Conference of Open Innovations Association (FRUCT)*, 193–199. doi:<ALIGNMENT.qj></ALIGNMENT>10.23919/FRUCT49677.2020.9211057
- Schroff, F., Kalenichenko, D., & Philbin, J. (2015). *FaceNet: A Unified Embedding for Face Recognition and Clustering*. https://openaccess.thecvf.com/content_cvpr_2015/html/Schroff_FaceNet_A_Unified_2015_CVPR_paper.html
- Serra, G., Ara, G., Fara, P., & Cucinotta, T. (2020). An Architecture for Declarative Real-Time Scheduling on Linux. *2020 IEEE 23rd International Symposium on Real-Time Distributed Computing (ISORC)*, 20–28. doi:<ALIGNMENT.qj></ALIGNMENT>10.1109/ISORC49007.2020.00013
- Swain, M., Singh, R., & Hashmi, Md. F., & Gehlot, A. (2020). Performance Analysis of Various Embedded Linux Firmwares for ARM Architecture Based IoT Devices. In G. Singh Tomar, N. S. Chaudhari, J. L. V. Barbosa, & M. K. Aghwariya (Eds.), *International Conference on Intelligent Computing and Smart Communication 2019* (pp. 1451–1460). Springer. doi:<ALIGNMENT.qj></ALIGNMENT>10.1007/978-981-15-0633-8_143
- Vishnubhatla, A. (2020). Linux Server Based Automatic Online Ticketing Kiosk. In A. P. Pandian, R. Palanisamy, & K. Ntalianis (Eds.), *Proceeding of the International Conference on Computer Networks, Big Data and IoT (ICCBI - 2019)* (pp. 428–439). Springer International Publishing. doi:<ALIGNMENT.qj></ALIGNMENT>10.1007/978-3-030-43192-1_49
- Zhang, S., Meng, X., Wang, L., & Liu, G. (2017). Research on Linux Kernel Version Diversity for Precise Memory Analysis. In B. Zou, M. Li, H. Wang, X. Song, W. Xie, & Z. Lu (Eds.), *Data Science* (pp. 373–385). Springer. doi:<ALIGNMENT.qj></ALIGNMENT>10.1007/978-981-10-6385-5_32

A. S. M. 梅赫迪·哈桑·萨德于 2020 年在美国国际大学 - 孟加拉国获得电气与电子工程学士学位。他的研究兴趣包括电路设计、计算机视觉、智能系统、物联网、嵌入式系统、聚合物基器件、电子学、自动化和机器人技术。目前，他在一家半导体公司担任电路与系统设计工程师。

Md Mashrur Sakib Choyon 于 2020 年在美国国际大学 - 孟加拉国获得电气与电子工程学士学位。他是电气与电子工程师协会(IEEE)的学生会员。他的研究兴趣包括：智能系统、物联网、电路设计、嵌入式系统、聚合物基器件、电子学、自动化和机器人技术。

阿布·哈斯纳特·穆罕默德·里德万于 2020 年在美国国际大学(孟加拉国)获得电气与电子工程学士学位。他的研究兴趣在于：智能电网、可再生能源和电力电子学。

卡维什克·希克德目前在美国国际大学 - 孟加拉国(AIUB)工程学院电气与电子工程系担任助理教授。他于 2015 年在 AIUB 获得电气与电子工程硕士学位(M.Sc in EEE),并以“最优等”成绩获得金质奖章。2013 年,他在同一所大学获得电气与电子工程学士学位(B.Sc in EEE)。希克德先生迄今已发表 4 篇期刊论文、13 篇会议论文,还有几篇已提交给不同的期刊和会议进行评审。他的主要研究方向是半导体材料、可穿戴设备材料、高速/高带宽设备材料、纳米制造、纳米电子学和光电子器件。

乔杜里·阿克拉姆·侯赛因分别于 2008 年和 2010 年在美国国际大学 - 孟加拉国(AIUB)达卡校区获得电气与电子工程(EEE)学士学位和电信工程硕士学位(以最优等成绩毕业)。目前,他在马来西亚苏丹再纳阿比丁大学攻读远程医疗博士学位。2010 年,他加入美国国际大学 - 孟加拉国,担任学生事务办公室的特别助理以及工程学院电气与电子工程系的高级讲师,目前他担任该系的高级助理教授。他的研究兴趣包括电力工程、无线通信、可再生能源、智能电网、自组织网络等领域。除了是 IEEE 高级会员外,他过去曾担任 VOLT 导师,目前是 IEEE MGA 培训委员会的成员。