Embedded Technoloy and Intelligent Systems 嵌入式技术与智能系统, 2025, 2(1), 41-47
Published Online February 2025 in Hans. https://www.hanspub.org/journal/etis
https://doi.org/10.12677/etis.2025.21004

嵌入式平台物体追踪应用实例：人脸追踪风扇

张　岩*，许　鹏，宋　岩

恩智浦半导体有限公司，北京

收稿日期：2025年1月20日；录用日期：2025年2月14日；发布日期：2025年2月28日

摘　要

本文聚焦基于MCU平台的物体检测算法的讨论，并提供了一套完整的工程实现示例：人脸追踪风扇。工程不仅提供了一个高效的物体检测算法，并且实现了一个自动追踪人体的控制系统，控制双路舵机转动底座，实现了风扇一直跟随人脸转动。本文所提供的检测控制系统也可以集成到监控设备，智能家居，工业自动化领域等等。

关键词

微控制器单元(MCU)，神经网络处理器(NPU)，物体追踪，人工智能应用

Embedded Platform Object Tracking Example: Face-Tracking Fan

Yan Zhang*, Peng Xu, Yan Song

NXP Semiconductors Limited, Beijing

Received: Jan. 20th, 2025; accepted: Feb. 14th, 2025; published: Feb. 28th, 2025

Abstract

The article focuses on the discussion of object detection algorithms based on the MCU platform and provides a complete set of engineering implementation examples: the face-tracking fan. This project not only provides an efficient object detection algorithm but also realizes a control system that can automatically track the human body. It controls the rotation of the base through two servos, enabling the fan to keep rotating following the human body. The detection and control system provided in the article can also be integrated into monitoring equipment, smart homes, industrial automation fields, and so on.

*通讯作者。

文章引用: 张岩, 许鹏, 宋岩. 嵌入式平台物体追踪应用实例: 人脸追踪风扇[J]. 嵌入式技术与智能系统, 2025, 2(1): 41-47. DOI: 10.12677/etis.2025.21004

Keywords

Microcontroller Unit (MCU), Neural Process Unit, Object Tracking, Artificial Intelligence Application

1. 引言

近年来，嵌入式平台物体追踪技术呈现出蓬勃发展的态势，在诸多领域展现出了强大的应用价值，并随着科技进步持续拓展其应用边界与深度，不断推动各行业朝着智能化方向迈进。

在安防领域，嵌入式平台物体追踪技术已然成为安全防范工作的得力助手。借助各类传感器与高清摄像头，能够精准捕捉可疑人员、车辆等目标的实时动态，实现对其全方位、不间断的追踪。例如，在大型商场、机场等人流量密集且安全要求较高的场所，监控系统中的嵌入式设备可以通过物体追踪技术，迅速锁定行为异常的人员，跟踪其行动轨迹，一旦发现有违法或危险行为的迹象，便能及时通知安保人员进行干预，有效预防盗窃、恐怖袭击等安全事件的发生，极大地提升了区域的整体安全性。

工业自动化领域同样受益于嵌入式平台物体追踪技术。在现代化的生产线上，该技术可对产品从原材料加工到成品组装的各个环节进行精确跟踪。通过实时监测产品的位置、状态等信息，确保每一道工序都能准确无误地执行，不仅有助于提高生产效率，避免因人为疏忽或机器故障导致的生产延误，还能在质量把控方面发挥关键作用，及时发现生产过程中的次品或缺陷产品，方便企业及时采取纠正措施，保障产品质量的稳定性。

智能交通方面，嵌入式平台物体追踪技术更是不可或缺。它可以通过安装在道路关键位置的摄像头以及车载终端等设备，实时追踪车辆的行驶轨迹、速度等信息。交通管理部门利用这些数据，能够实现对交通流量的智能调控，比如根据实时路况动态调整信号灯时长，合理引导车辆分流，缓解交通拥堵状况；同时，对于超速、违规变道等交通违法行为也能做到精准抓拍，维护良好的交通秩序，提升道路交通安全水平。

智能家居场景下，嵌入式平台物体追踪技术为人们的生活带来了诸多便利与舒适体验。例如，通过追踪家庭成员在室内的活动情况，智能照明系统可以实现人来灯亮、人走灯灭的自动化控制；空调系统能依据人员所在位置及活动状态，自动调节室内不同区域的温度，达到节能且舒适的效果；智能电视等娱乐设备也可以根据用户的观看习惯和位置，自动调整画面角度和音量大小等参数，为用户打造个性化、智能化的家居环境[1]。

随着 MCU 产品的迅猛发展与快速迭代，MCU 早已突破传统控制单元的局限。如今，它的计算能力与资源配置已得到极大提升，足以支撑更为繁杂的运算任务。MCU 凭借着低成本、低功耗的显著优势，以及与生俱来的卓越实时性，为实时控制系统注入了人工智能的强大活力，使其成功具备 AI 功能。

本文以一个能追踪人脸位置的风扇为例，添加了物体检测追踪和电机控制模块，使得风扇具备了实时的人脸追踪功能。

2. 工程实现

人脸追踪风扇作为嵌入式平台物体追踪技术的一个具体应用案例，有机地将多种先进技术整合在一

起，实现了风扇对人脸的实时追踪功能。

首先，主控芯片的选择至关重要，本案例选用了恩智浦的 MCXN947 芯片，它搭载了双核 Cortex-M33 作为主运算单元，主频高达 150 M，能够满足复杂运算对处理速度的要求。同时，内置了 512 K 的 RAM 和 2 M 的 Flash，为数据的临时存储和程序的运行提供了必要的空间支持。尤为关键的是，芯片集成了一颗 NPU (神经网络处理单元)，这一组件在加速运行神经网络模型方面发挥着不可替代的作用，使得物体追踪算法能够在嵌入式平台有限的资源条件下得以高效执行，为整个系统的稳定运行和高效运算提供了坚实基础。

在图像采集环节，摄像头负责采集外界的图像信息，将其作为后续处理的原始数据输入。采集到的图像分辨率为 640 × 480，数据格式为 RGB565，不过一帧数据大小达到了 600 KB，超出了 MCXN947 内部 RAM 的大小。为解决这一数据处理难题，系统运用了 SmartDMA 技术对图像数据进行切片处理，每次中断仅处理部分数据，并且把接收到的原始图像经过剪裁后显示在 LCD 上，方便用户直观查看实时图像；另一方面，我们把原始图像经过下采样后，并且把下采样后的切片数据增量缓存，直到接收完一帧图像的所有切片，得到完整的下采样后的图像，再将其输入到物体追踪算法中进行进一步分析。

算法运算阶段是实现人脸追踪功能的核心环节。我们先是通过物体检测算法对输入的图像数据进行分析，计算得出物体的位置信息，具体而言，就是能够确定人脸的中心点坐标以及长、宽等关键参数。随后，在 LCD 上绘制白色框，将其叠加显示在实时图像上，这样使用者便可以直观地看到算法的运算结果。接着，对获取到的位置信息进行 SORT (Simple Online and Realtime Track)运算，通过该运算可以得到稳定的轨迹信息，为后续电机控制提供精准的数据依据。

最后是电机控制环节，基于前面算法运算所得到的轨迹信息，通过 PWM (脉冲宽度调制)技术来控制水平和垂直两个方向的舵机进行相应转动，进而带动风扇旋转，使其能够精准地追踪摄像头采集到的人脸位置，确保风扇始终朝着人脸所在方向送风，实现了人脸追踪风扇这一智能化的功能应用。系统结构见图 1 所示。

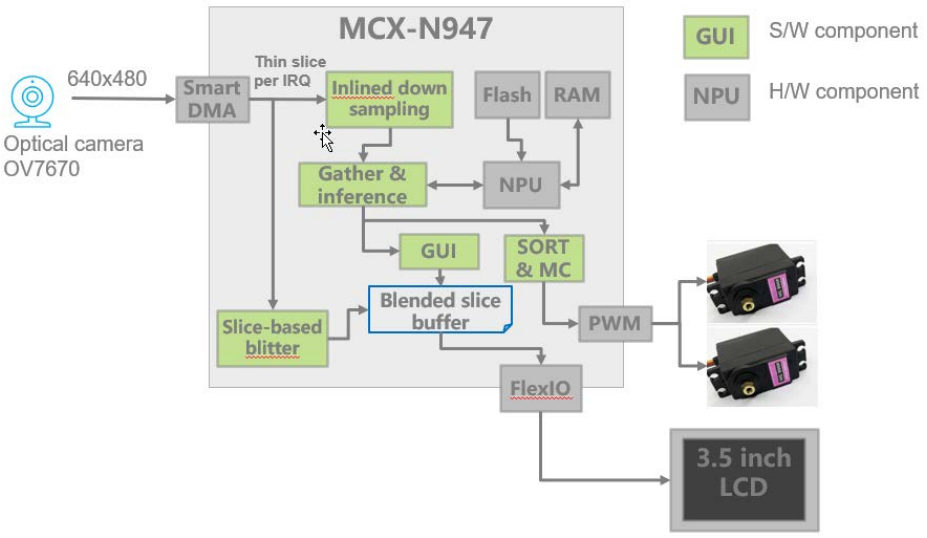

Figure 1. Block diagram of system
图 1. 系统框架图

3. 物体追踪算法

目前常用的物体追踪算法主要有：传统算法和基于深度学习的神经网络算法。

传统算法常用帧差法和均值漂移法。帧差法[2]通过比较相邻两帧图像之间的像素差异来判断物体是否发生移动，进而实现对物体的追踪。其核心在于，移动的物体在连续的视频帧中会呈现出明显的像素变化，而静止的背景像素变化则相对较小。它的计算过程较为简单，对硬件计算资源的要求相对不高，这使得其能够很好地适配那些计算能力有限的嵌入式设备。同时它局限性也很明显。它对光照条件的变化较为敏感，当光照出现明显改变时，比如从白天到夜晚、或是有强光突然照射等情况，即使物体本身没有移动，前后帧的像素差异也可能会很大，从而容易导致误判，影响追踪的准确性。同时，对于物体的阴影部分，帧差法可能会将其误识别为移动物体，而且在面对快速移动的物体时，由于帧率的限制，可能出现漏检的情况，像快速奔跑的小动物等在画面中可能无法被准确追踪。

均值漂移法[3]是一种基于密度估计的非参数算法，其算法流程大致如下：首先，在图像中选择一个初始窗口(通常为矩形)，这个窗口用于框定要追踪的目标物体；接着，计算该窗口内像素的颜色直方图，以此作为目标物体的模型特征表示；然后，根据目标模型的颜色分布情况，在后续的帧图像中，通过不断迭代计算窗口的平均漂移量，也就是计算当前窗口内像素的颜色直方图与目标模型颜色直方图之间的差异，进而更新窗口的位置和尺寸，最终实现目标的跟踪。它的局限性是无法处理背景复杂，相对特征不明显的物体检测。

随着深度学习的发展，由基于神经网络的物体检测算法加上基于卡尔曼滤波实现的 SORT [4]算法逐渐成为物体追踪的主流实现方法，这也是我们采用的方法，接下来详细介绍。

3.1. 物体检测算法

目前在嵌入式平台上比较成熟的基于深度学习的物体检测算法主要由两大类：SSD 算法以及 YOLO 算法。

3.1.1. 模型选择

SSD (Single Shot MultiBox Detector) [5]算法同样是一种基于深度学习的目标检测算法，它属于单阶段(one-stage)检测方法，能够在单次前向传播中就完成目标检测任务，兼具速度快和精度相对较高的特点。SSD 的一个显著区别在于它提取了不同尺度的特征图来做检测，大尺度特征图(较靠前的特征图)可以用来检测小物体，而小尺度特征图(较靠后的特征图)用来检测大物体，这使得它在对不同大小目标的检测上更加全面和准确。

YOLO (You Only Look Once) [6]系列算法在嵌入式设备上的物体追踪应用中有着显著的特点。其核心思想是将物体检测问题转换为回归问题，采用单个神经网络直接在整个图像上预测边界框和类别概率，摒弃了传统的基于区域建议的复杂流程，实现了端到端的检测。在嵌入式设备这种对实时性要求较高、计算资源相对受限的环境下，YOLO 算法的快速检测和识别目标能力凸显出极大优势。

YOLO 与 SSD 相比，精度更高且小目标识别率更好，YOLO 模型的网络结构更加复杂，检测头网络具备特征融合机制，可以更好的利用全局上下文信息提升检测效果，在遮挡或复杂背景下表现更好。在嵌入式设备上选择物体追踪算法时，需要综合考量多方面的因素。首先硬件资源是一个重要的考量因素，过大的模型无法适配到 MCU 平台上。其次计算资源方面，不同的模型算子在 NPU 上的支持与否也很重要。YOLO V3 版本算子简单更适合部署在 NPU 上运行，而且其实现算法简单，更适剪裁优化。

3.1.2. 模型优化

标准 YOLO V3 Tiny 版本模型经过训练后权重有 8 M 左右，运行内存需求 6 M，远远超出了 MCXN947 的存储大小。所以需要我们对模型的网络结构进行剪裁。

原始网络结构由骨干网络 Darknet53 加检测头 FPN 组合而成，结构图如下图 2 所示。

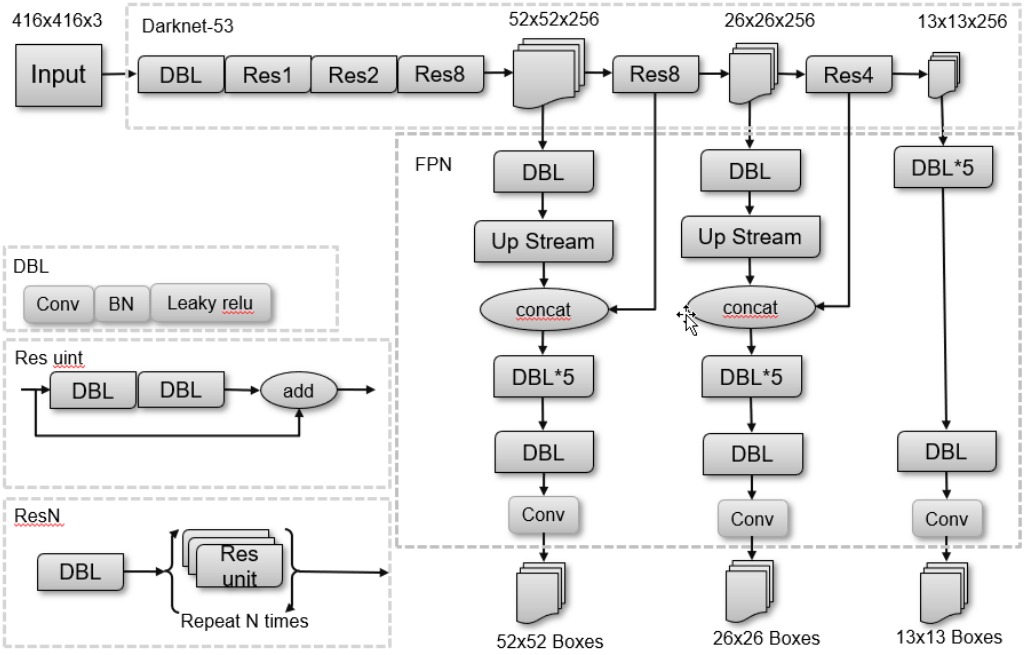

Figure 2. YOLO network
图 2. YOLO 网络结构图

优化方向从两个方面入手：首先缩减输入分辨率，由 416 × 416 缩减至 128 × 160。然后缩减骨干网络，改用 MobileNet 网络结构，见图 3。

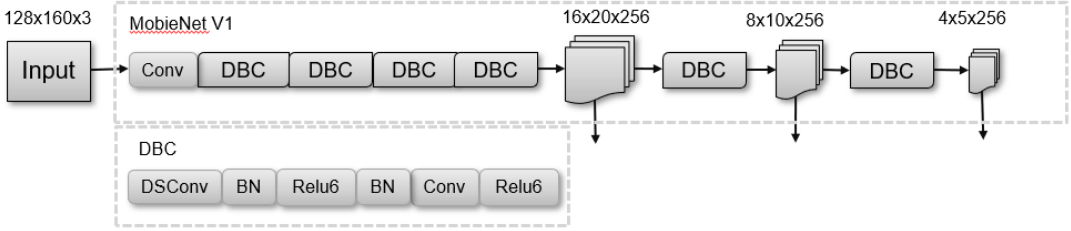

Figure 3. Backbone network
图 3. 骨干网络

从图 2 可知 Darknet-53 作为骨干网络共有 52 个卷积，并且大量使用了残差链接。残差链接对内存的需求量很大，不利于在 MCU 平台上部署。改用 MobileNet 极大减少了内存的使用，同时支持预训练权重，缩短了训练时间。骨干网络分别提取 1/8 大小的特征图，1/16 大小特征图以及 1/32 大小特征图。

检测头网络 FPN 根据根据骨干网络提取的特征值进行进一步提炼与融合，最终输出 3 个尺度的目标预测，结构见图 4。

经过优化后，模型大小为 280 KB，运行 RAM 需求是 195 KB，量化后在 NPU 上运行时间为 22 ms，可以满足实时控制需求。

3.2. 追踪算法

经过模型运算后得到人脸的二维坐标信息，就可以控制电机根据人脸位置进行旋转。接下来有两个

问题需要解决。第一个问题：模型结果会有误差会导致物体中心点位置出现抖动，需要引入算法解决抖动问题。第二个问题：模型检测到多个物体，怎么判断当前该跟随哪个目标移动。

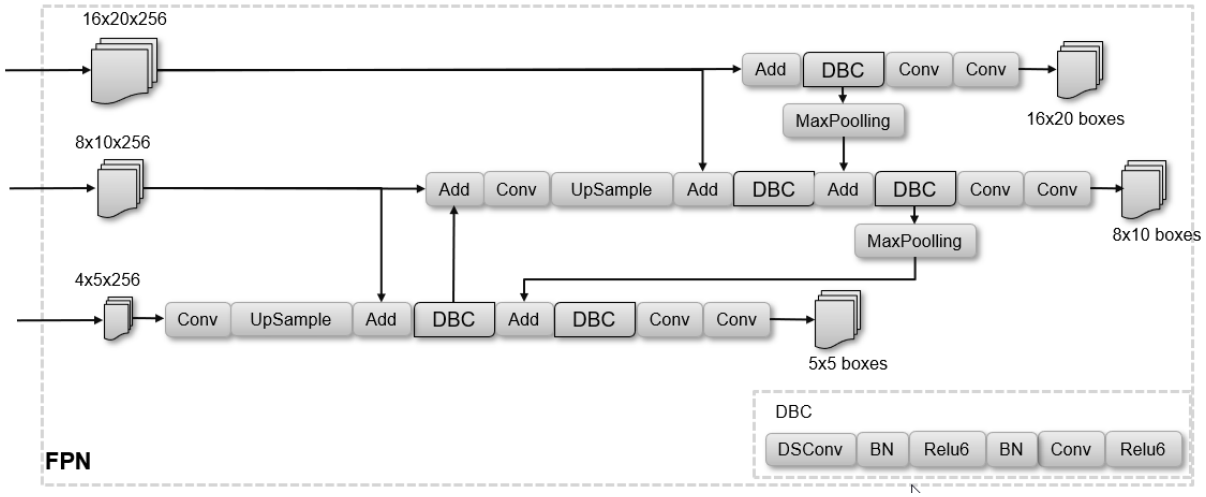

Figure 4. FPN network
图 4. 检测头网络

SORT 物体追踪算法可以解决以上两个问题。SORT 算法主要基于物体的运动模型进行追踪，对于运动比较规律的目标能够取得较好的追踪效果，其计算复杂度较低，能够满足实时性要求。广泛应用于视频监控，自动驾驶以及机器人视觉等领域。算法流程见图 5。

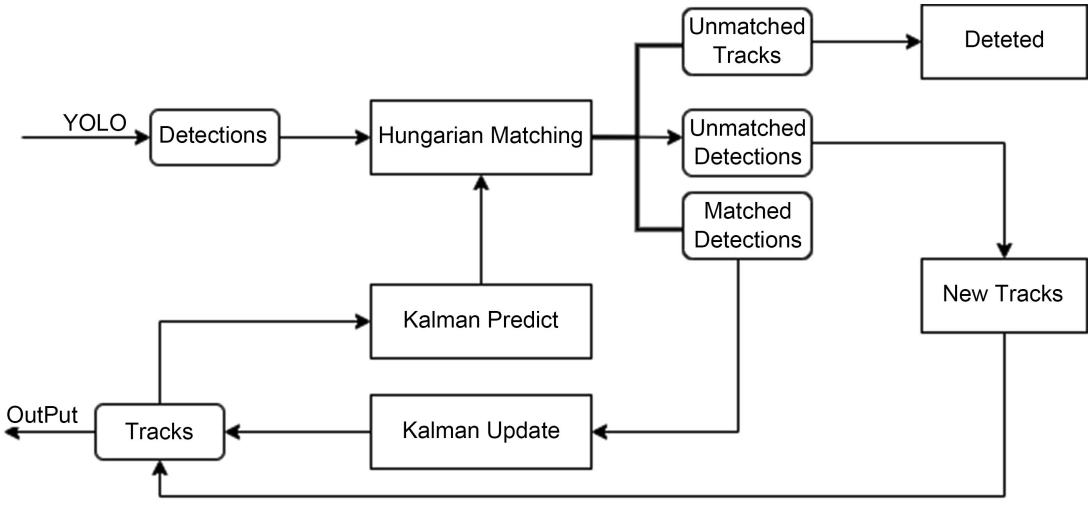

Figure 5. Algorithm workflow
图 5. 算法流程图

从上图所示，YOLO 模型检测得出多个目标位置信息，经过匈牙利匹配算法，查找历史记录中是否有与当前目标匹配的记录，发现匹配记录则更新卡尔曼滤波器中轨迹信息及协方差矩阵。未发现匹配记录表示出现新目标，根据新目标的位置信息新建卡尔曼滤波器实例。同时遍历其它滤波器，如果没有新的轨迹更新则表示目标脱离，删除无效滤波器。当所有滤波器更新完毕后根据更新后的轨迹控制电机，追踪目标。

4. 实验验证

基于 WIDER-FACE [7]生成人脸检测训练集，原始数据集图像尺寸过大，大部分图片包含身体部分，不适用当前的应用场景，需要根据模型输入大小优化精简数据集。使用 YOLOV3 训练脚本训练模型，在训练开始阶段加载 MobileNetV1 预训练权重可以提高训练效率的同时提高模型精度。经过 50 轮训练后，平均精度 mAP = 89.78%。

将训练后的模型进行量化，保存成 INT8 数据格式的 tflite 文件。使用恩智浦官网提供的 eIQ Toolkit 将模型转换成 NPU 可以识别的数据类型，部署在硬件 MCX-N947。系统运行时检测结果实时显示在 LCD 上，可以看到模型检测人脸的消耗时间是 24 ms，可以满足实时性控制需求。

5. 更多讨论

本文实现了一种能够在资源有限的微控制器单元(MCU)上运行的物体追踪算法。还有很多细节问题可以进一步讨论，比如：

环境适应性挑战。光照条件的变化对嵌入式设备上物体追踪的准确性有着显著影响，在强光直射的情况下，物体表面可能会出现过曝现象，原本的纹理、颜色等特征变得模糊不清，这使得基于特征识别的追踪算法难以准确提取有效的特征信息，从而导致追踪失败或者追踪位置出现较大偏差。而从白天到夜晚的光照变化过程中，光线逐渐变暗，物体的对比度降低，对于一些依靠颜色、灰度等特征来区分物体和背景的追踪算法来说，其区分度下降，误判的概率大幅增加。可以引入功能更强的摄像头或引入补光设备降低光照对输入图像的影响。

再比如遮挡问题，在实际应用场景中，物体被遮挡是经常出现的情况，这给物体追踪带来了很大的困难。当物体被部分遮挡时，其可见的特征信息减少，追踪算法依据有限的特征进行匹配和定位的难度增大，容易出现误跟或者跟丢的现象。可以引入多摄像头不同角度采集图像解决遮挡问题。

目前物体追踪主要依赖视觉数据居多，未来结合其他模态数据如声音、红外热成像、雷达信号等进行融合追踪将是重要发展方向。不同模态的数据可以相互补充、验证，在复杂环境下提高追踪的准确性和鲁棒性。例如，在低光照、烟雾等视觉受限的场景中，利用红外热成像和声音信息辅助。

文中配合使用的代码已经开源，地址：https://github.com/nxp-appcodehub/dm-multiple-face-detection-on-mcxn947。

参考文献

[1] Kumar, A., Jain, R., Vairamani, A.D. and Nayyar, A. (2023) Object Tracking Technology: Trends, Challenges and Applications. Springer.

[2] Rao C.S., and Darwin, P. (2012) Frame Difference and Kalman Filter Techniques for Detection of Moving Vehicles in Video Surveillance. https://www.ijera.com/papers/Vol2_issue6/FN2611681170.pdf

[3] Bandung, Y. and Ardiansyah, A. (2021) Mean-Shift Object Tracking Algorithm with Systematic Sampling Technique. https://scholarhub.ui.ac.id/mjt/vol25/iss1/4/

[4] Bewley, A., Ge, Z.Y., Ott, L., Ramos, F. and Upcroft, B. (2016) Simple Online and Realtime Tracking. https://arxiv.org/abs/1602.00763

[5] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. and Berg, A.C. (2015) SSD: Single Shot MultiBox Detector. https://arxiv.org/abs/1512.02325

[6] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2015) You Only Look Once: Unified, Real-Time Object Detection. https://arxiv.org/abs/1506.02640

[7] WIDER FACE: A Face Detection Benchmark. http://shuoyang1213.me/WIDERFACE/