团 体 标 准

T/BITS 0036-2025

城市主要交通节点车-步协同监测 系统技术规范

2025-06-30 发布

2025-07-30 实施

保定市智能交通学会 发布

目 次

1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 系统组成	2
4.1 硬件部分	2
4.2 软件部分	2
5 技术要求	2
5.1 车流量检测	2
5.2 行人过街行为检测	3
5.3 设备部署与信息采集	3
5.4 数据处理与上报	3
6 信息安全要求	3
6.1 基本信息安全功能	3
6.2 安全功能具体需求	4
7 性能指标	4
7.1 车流量检测性能指标	4
7.2 行人过街行为检测性能指标	4
7.3 数据传输与上报性能指标	4
8 指标参数	4
8.1 相机主要参数	4
8.2 系统参数	5
8.3 算法参数	5
9 测试与验证	5
9.1 测试目的	5
9.2 测试范围	5
9.3 测试类型	5
9.4 测试方法	6
9.5 测试环境与工具	6
9.6 验证流程	6
9.7 问题处理与验证	7
10 维护与管理	7
10.1 系统维护与管理	7
10.2 网络维护与管理	7
10.3 硬件维护与管理	7
10.4 软件维护与管理	7

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由保定市智能交通学会提出并归口。

本文件起草单位：保定维特瑞光电能源科技有限公司、维特瑞交通科技有限公司、维特瑞交通科技（河北）有限公司、天响创新（河北）科技有限公司、河北大中交通规划设计有限责任公司、喆凯信息技术（河北）有限公司、河北勃迩蒂信息科技有限公司、保定玥合信息技术有限公司、河北雄业华阳交通科技有限公司、智控云检测技术（河北）有限公司、天堡航空科技（河北）有限公司、仁礼科技教育发展（河北）有限公司。

本文件主要起草人：王士元、王川、戈晶宇。

城市主要交通节点车-步协同监测系统技术规范

1 范围

本文件规定了城市主要交通节点车-步协同监测系统技术规范，包括系统组成、技术要求、信息安全要求、性能指标、指标参数、测试与验证及维护与管理等内容。

本文件适用于城市主要交通节点车-步协同检测，其他路口参照执行。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

- GB/T 15566 公共信息导向系统
- GB/T 24405.1-2009 信息技术服务管理 第1部分:规范
- GB/T 28181-2011 安全防范视频监控联网系统信息传输、交换、控制技术要求
- GB/T 28827.1 信息技术服务运行维护 第1部分:通用要求
- GB/T 28827.2 信息技术服务运行维护 第2部分:交付规范
- GB/T 28827.3 信息技术服务运行维护 第3部分:应急响应规范
- GB/T 28827.6 信息技术服务运行维护 第6部分:应用系统服务要求
- GB/T 30285-2013 信息安全技术灾难恢复中心建设与运维管理规范
- GB/T 33136-2016 信息技术服务数据中心服务能力成熟度模型
- GB/T38633-2020 信息技术大数据系统运维和管理功能要求
- GB 50174-2017 数据中心设计规范
- GB 50348-2004 安全防范工程技术规范
- GB 50259-92 电器装置安装工程电器照明装置施工及验收规范
- GA/T 367-2001 视频安防监控系统技术要求
- GA887-2010 中华人民共和国公共安全行业标准
- GA-T 751-2008 视频图像文字标注标准
- GA/T 1043 道路交通技术监控设备运行维护规范

3 术语和定义

3.1 术语和定义

下列术语和定义适用于本文件。

3.1.1 城市交通节点 complex urban intersections

指城市中由多条道路交叉形成的，具有特殊交通组织需求和较高交通管理难度的路口，通常由于道路条件、交通流量、交通参与者多样性等因素而显得复杂。

3.1.2 车流量检测 traffic flow detection

对通过指定路口的车辆数量、车辆类型等信息的检测。

3.1.3 行人过街行为检测 pedestrian crossing behavior detection

利用视觉技术来判断和识别行人在过马路时的行为，涉及到对图像或视频序列中行人动态的分析，确定行人是否正在穿越道路，进一步评估其过街行为的合规性和安全性。

4 系统组成

城市交通节点车流量及行人过街行为检测系统应包括硬件部分和软件部分，硬件部分包括工业相机、交换机、路由器、工控机及其他辅助设备；软件部分包括图像识别、数据处理及调试工具等。

4.1 硬件部分

4.1.1 工业相机应具备高清图像采集和传输功能，并支持远程控制和调节。

4.1.2 交换机应具备高速数据传输并具备良好的扩展性，应能实现大规模的网络部署。

4.1.3 路由器应支持多种路由协议，如 OSPF、BGP 等，应具备强大的安全功能，如防火墙、VPN、入侵检测等。

4.1.4 工控机作为系统核心，应具备数据处理和控制能力、通信与联网功能、监测与数据采集能力、设备控制和调度能力、扩展性以及其它特性。

4.1.5 其他辅助设备可包括存储设备、电源设备等。

4.2 软件部分

4.2.1 图像识别

- a) 应能够实时检测视频流中的车辆，并准确识别车辆的位置和轮廓；
- b) 应能根据车辆的外观特征区分车辆类型，如轿车、货车、公交车等；
- c) 应支持车牌识别功能，能够自动从车辆图像中提取车牌号码，并进行准确的识别；
- d) 应能检测视频流中的行人，并准确识别行人的位置和姿态；
- e) 应能识别视频流中的非机动车，如自行车、电动车等，并准确判断其行驶方向和速度；

4.2.2 数据处理

- a) 应能够统计指定时间段内通过路口的车辆数量、车辆类型等信息，并生成相应报表；
- b) 应能够分析交通流量和车辆行驶速度，判断是否存在交通拥堵情况，并及时发出预警；
- c) 通过分析车辆行驶轨迹、速度、加速度等参数，应能识别交通事故，如追尾、碰撞等，并触发报警机制；
- d) 通过对行人、非机动车的过街行为进行分析，应能够识别不安全的过街行为，如闯红灯、逆行等，并为交通管理部门提供数据支持。

4.2.3 调试工具

用于相机的标定和参数设置，应支持远程调试和监控。

5 技术要求

5.1 车流量检测

5.1.1 系统应能实现城市交通节点车道级车流量的实时检测，支持对多个车道进行并行检测，支持多个车道的车辆流量信息同步处理。

5.1.2 相机需具备高分辨率图像捕捉能力，能够清晰捕捉车辆图像，支持图像识别算法进行车辆分类和计数。

5.1.3 能够对车辆类型（如小型车、大型车、货车等）进行分类，并对车辆数量进行自动计数，准确统计各车道内的各类车辆数量及车流量信息。

5.2 行人过街行为检测

5.2.1 系统应实现城市交通节点人行横道及行人/非机动车过街行为的检测，能够准确判断并记录过街行为，包括是否遵守交通规则、是否在人行横道内行走等。

5.2.2 相机应具备角度调节功能，调整至适当角度覆盖人行横道区域，支持行人/非机动车的识别和跟踪。

5.2.3 系统应能够识别并适应不同的过街场景，如人行横道、交通信号灯控制的过街路口、无信号灯控制的过街点等，应能在各种情况下准确判断并记录过街行为。

5.2.4 当系统检测到行人/非机动车违规过街存在潜在危险时，应能够实时发出警报，并记录相关视频和图像作为证据，以便后续处理和管理。

5.3 设备部署与信息采集

5.3.1 应实现单点相机对指定车道进行部署，相机可与交通信号灯等设施集成于一体，确保相机视角覆盖指定车道主体区域。

5.3.2 相机应支持实时数据传输功能，确保车道级信息（包括车辆图像、分类、车流量等）能够实时、准确地传输至交通管理中心。

5.3.3 相机应支持远程监控和参数调整功能，交通管理部门可通过远程控制系统实时查看相机画面，并根据需要对相机的拍摄角度、图像质量等参数进行调整。

5.3.4 相机应配备夜视功能，确保在夜间或低光环境下也能清晰捕捉车辆图像，同时应具备抗干扰能力，防止因恶劣天气、光线变化或外界干扰等功能。

5.4 数据处理与上报

5.4.1 应实现对各车道信息的算法处理，包括车辆分类、计数、流量统计等。

5.4.2 处理结果应通过 IP 地址和 485 总线进行实时上报与定时上报，支持多种上报模式和接收端配置。

5.4.3 在数据传输过程中，应严格实施数据安全保障措施，采用加密技术对上报数据进行加密处理，确保数据在传输过程中不被非法截获和篡改。

5.4.4 系统应具备数据质量监控与校验机制，对于异常数据或不符合预期的结果，系统应能自动进行识别和标记，还应定期进行数据校验，确保上报数据的准确性和可靠性。

6 信息安全要求

6.1 基本信息安全功能

6.1.1 安全功能应能够灵活部署与扩展。

6.1.2 应能够在一定时间内持续抵抗顶击。

6.1.3 应能够允许一定程度和范围内的功能失效，但基础功能始终保持运行。

- 6.1.4 整个系统应能够从失败中快速完全恢复。
- 6.1.5 安全功能应轻量化设计，能够部署在各类硬件资源受限的物联网设备中。
- 6.1.6 关键设备应设置网络隔离，控制安全攻击和风险范围，避免攻击范围扩大。
- 6.1.7 应设置统一的态势感知、安全管理与编排、统一的身份认证与管理，以及统一的安全运维体系，保障整个系统安全与可靠。

6.2 安全功能具体需求

- 6.2.1 应具备远程升级及修复漏洞和补丁的功能，同时避免升级后系统失效。
- 6.2.2 应具备防火墙、入侵检测和防护、VPN/TLS 功能，结合深度定制的安卓、北斗等安全系统，重要数据通过公安专网传输至交通管理部门指挥中心，其他信息推送至政务云平台。
- 6.2.3 应具备数据加密、数据隔离和销毁、数据防篡改、隐私保护（数据脱敏）、数据访问控制和数据防泄漏等功能，特别要考虑到分布式部署的设备被非法侵入及窃取后，相关运行数据不会泄露，紧急情况下具有内部数据自删除及锁定功能。
- 6.2.4 应包含白名单、应用安全审计、恶意代码防范、WAF（Web 应用防火墙）等安全功能，设置访问权限，防止异常业务接入。
- 6.2.5 应具备身份和认证管理功能，还可考虑去中心化、分布式的认证方式和证书管理。
- 6.2.6 可设置基于大数据的态势感知和高级威胁检测，统一的全网安全策略执行和主动防护，结合完善的运维监控和应急响应机制，最大限度保障系统的安全可信。

7 性能指标

7.1 车流量检测性能指标

- 7.1.1 系统每秒检测车辆数量不宜少于 30 辆。
- 7.1.2 车辆数量检测的准确率应达到 98%以上，车辆分类的准确率应达到 95%以上。
- 7.1.3 从检测到数据上报的延迟不宜超过 1 秒。

7.2 行人过街行为检测性能指标

- 7.2.1 系统应能在距离人行横道 5-10 米的范围内准确识别行人及非机动车过街行为。
- 7.2.2 漏检率应低于 3%。
- 7.2.3 在光照强度变化范围-10lux~10000lux 内，系统仍能准确检测并识别过街行为。

7.3 数据传输与上报性能指标

- 7.3.1 IP 地址传输速度不低于 100Mbps，485 总线传输速度不低于 9.6Kbps。
- 7.3.2 数据传输过程中，数据丢失率应低于 0.1%。
- 7.3.3 支持实时上报和定时上报两种方式，定时上报时间间隔可设置。

8 指标参数

8.1 相机主要参数

- 8.1.1 系统宜采用嵌入式 RTOS 设计，双核 32 位 DSP(GK7205V300)，纯硬压缩，并配备看门狗。
- 8.1.2 采用至少达到 500 万像素的 1/2.8"SC5239S 低照度 CMOS 传感器；彩色模式下，最低照度达到 0.01Lux@F1.2；黑白模式下，最低照度达到 0.001Lux@F1.2。

8.1.3 视频处理与编码应支持 H.265+（兼容 H.265/H.264）编码标准；应支持双码流输出，满足不同的应用需求；应支持 AVI 格式视频输出；码流范围可调，从 0.1M~8Mbps；帧率 1~30 帧/秒可调，保证实时性和流畅性。

8.1.4 图像输出主码流支持多种分辨率和帧率，包括 2880*1616@25fps，2560*1296@fps，2304*1296@25fps，1920*1080@25fps；子码流应支持 800*448@25fps。

8.1.5 支持 WEB 配置、OSD（屏幕显示叠加）、移动侦测；移动侦测报警后中心提醒，画面弹出联动；支持 FTP 抓图功能；支持客户端远程监控软件、MYEYE 平台等系统应用；提供完善的 SDK 开发包等业务。

8.1.6 网络接口至少应提供 1 个 RJ45 以太网接口，10/100M 自适应；支持 RTSP/FTP/DHCP/NTP/UPnP 等网络协议。

8.1.7 信号接口至少应提供 1 个光敏电阻接口，1 个 IR-CUT 驱动接口，支持光敏电阻输入信号与 IR-CUT, 红外灯的联动

8.2 系统参数

8.2.1 系统帧率应符合在 1 秒内处理至少 30 张车辆或行人/非机动车图像。

8.2.2 系统应具备足够的存储容量，能够存储至少 7 天的历史数据，内存占用率不应高于 70%。

8.2.3 系统应支持模块化设计，方便未来升级和扩展功能。

8.3 算法参数

8.3.1 车辆和行人/非机动车的识别速度应小于 0.5 秒/个。

8.3.2 算法应具有一定的学习能力，能够根据历史数据不断优化和提高识别准确率。

8.3.3 算法应具有较强的鲁棒性，能够应对各种复杂场景和天气条件下的识别任务。

9 测试与验证

9.1 测试目的

验证系统是否满足所有预定的功能要求和技术指标；评估系统在不同场景和条件下的性能和稳定性；发现和修复系统中存在的缺陷和错误；确保系统的易用性和可维护性。

9.2 测试范围

测试范围应覆盖系统所有关键功能和模块，包括但不限于：

- a) 车辆分类与计数算法；
- b) 流量统计与分析；
- c) 数据处理与上报机制；
- d) 用户界面与交互；
- e) 系统安全性与稳定性。

9.3 测试类型

测试类型可包括但不限于：

- a) 单元测试：针对单个模块或组件进行测试，确保其基本功能正确；
- b) 集成测试：将多个模块或组件集成在一起进行测试，验证交互和协作是否正常；
- c) 系统测试：对整个系统进行测试，验证其是否满足设计要求和相关标准；

- d) 验收测试：在系统部署到实际环境之前进行测试，确保系统能够满足用户的实际需求。
- e) 检查系统是否提供了必要的帮助文档和在线支持。

9.4 测试方法

9.4.1 功能测试

- a) 可将程序的输入域划分成若干部分（子集），从每一个子集中选取少数具有代表性的数据作为测试用例；
- b) 可针对输入或输出范围的边界设计测试用例；
- c) 可基于测试人员的经验，推测出容易出错的情况，并设计相应的测试用例；
- d) 可根据输入条件的组合、约束关系和输出条件的因果关系，分析输入条件的各种组合情况设计测试用例。

9.4.2 性能测试

- a) 可模拟多用户同时访问软件，测试软件在负载情况下的性能表现；
- b) 可在极端负载条件下测试软件的性能，如超出预期的用户访问量或数据量；
- c) 可将软件的性能与已知的基准进行比较，以评估软件的性能水平。

9.4.3 兼容性测试

- a) 可在不同操作系统上安装和运行软件，检查其功能和性能是否正常；
- b) 应确保软件在不同的Web浏览器上都能正确显示和运行；
- c) 应测试软件在不同硬件配置下的运行情况，如CPU、内存、硬盘等。

9.4.4 安全性测试

- a) 可模拟黑客攻击，尝试绕过软件的安全防线，以评估软件的安全性；
- b) 可使用专门的工具对软件进行扫描，发现潜在的安全漏洞；
- c) 验证软件中的权限管理功能是否有效，确保用户只能访问其被授权的资源。

9.4.5 回归测试

- a) 可使用自动化测试工具来执行回归测试，提高测试效率；
- b) 可根据修改的内容和风险，选择性地执行部分测试用例。

9.5 测试环境与工具

测试环境应尽可能模拟实际运行环境，包括硬件、软件、网络等方面。测试工具可根据测试类型和需求选择，如自动化测试工具、性能测试工具、安全测试工具、代码审查工具等。

测试工具宜选用：

- a) 自动化测试工具：如Selenium、Appium等，用于自动化执行测试用例和生成测试报告。
- b) 性能测试工具：如JMeter、LoadRunner等，用于模拟大量用户请求并收集性能数据。
- c) 安全测试工具：如OWASP Zap、Nmap等，用于发现系统中的安全漏洞。
- d) 代码审查工具：如SonarQube、Checkstyle等，用于辅助代码审查和质量控制。

9.6 验证流程

- a) 根据需求文档和技术规格书设计测试用例，包括正常情况和异常情况；
- b) 搭建与生产环境相似的测试环境，确保测试结果的准确性；
- c) 按照测试计划执行测试用例，并记录测试结果；
- d) 对测试过程中发现的缺陷进行记录、跟踪和修复；
- e) 根据测试结果编写测试报告，包括测试概述、测试结果、缺陷列表和测试建议等；
- f) 当所有测试用例均通过且满足预定的性能指标时，认为系统验证通过。

9.7 问题处理与验证

在测试过程中发现的问题应及时记录并报告给相关人员进行处理。问题处理完成后，应进行验证测试以确保问题已得到妥善解决。同时，应定期回顾和评估测试过程中发现的问题和解决方案，以改进测试流程和提高测试质量。

10 维护与管理

10.1 系统维护与管理

10.1.1 应定期对重要数据和系统文件进行备份，并建立快速有效的恢复机制，能够在系统出现故障时迅速恢复服务。

10.1.2 可通过性能监控工具，对系统资源使用情况、响应时间、吞吐量等关键指标进行实时监控。根据监控结果，进行性能优化，提升系统的运行效率。

10.1.3 应加强系统安全管理，设置合理的安全策略，定期更新安全补丁，加强用户权限管理。

10.2 网络维护与管理

10.2.1 应对网络设备（如路由器、交换机、防火墙等）进行定期检查、配置更新和故障排查，确保网络设备正常运行，为网络提供稳定的传输服务。

10.2.2 可通过网络性能监控工具，对网络带宽、延迟、丢包等关键指标进行实时监控。根据监控结果，进行网络优化，提升网络性能。

10.2.3 应加强网络安全管理，设置合理的安全策略，配置防火墙和入侵检测系统，加强用户访问控制。

10.3 硬件维护与管理

10.3.1 应定期对硬件设备进行巡检，及时发现潜在故障并进行排查。应对出现的硬件故障进行快速定位和处理，确保硬件设备的正常运行。

10.3.2 可通过硬件性能监控工具，对硬件设备的资源使用情况、性能瓶颈等进行分析 and 优化。

10.4 软件维护与管理

10.4.1 应及时修复软件故障、更新软件版本以修复已知的安全漏洞和错误。

10.4.2 可通过建立软件配置管理系统，对软件系统的配置信息进行集中管理，确保软件系统的可配置性和可维护性。

10.4.3 可通过软件测试确保软件系统的质量和稳定性，通过自动化部署工具实现软件系统的快速部署和升级。

10.4.4 应根据业务需求和技术发展，及时更新和升级硬件设备。