

智能相机管理系统的设计与实现

吴培浩, 李迪, 张成, 翁潇文

(华南理工大学机械与汽车工程学院, 广州 510640)

摘要: 针对传统的文本编程效率低下、智能相机只允许本地监视以及监视界面固定等问题, 自主研发了一套集图形化编程、任务运行、数据监控等功能于一体的智能相机管理系统; 分析了智能相机管理系统在编程、运行与监视过程中的功能需求, 并进行系统结构的划分和设计; 构建图形化图像处理单元、可视化编程界面以实现图形化编程功能; 采用网络通讯方式, 实现智能相机配置过程的同步、远程监控以及与其他设备的数据交互; 讨论了系统中的数据管理模型; 最后, 通过应用实例验证该系统在工业生产中的可行性; 验证结果表明, 该系统能缩短开发时间, 降低用户开发成本, 并能对生产线上的智能相机进行有效的管理。

关键词: 智能相机; 软件设计; 图形化编程; 界面组态

Design and Implementation of Management System of Smart Camera

Wu Peihao, Li Di, Zhang Cheng, Weng Xiaowen

(School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou 510640, China)

Abstract: For the shortage of low efficiency of traditional programming method and the problems that smart camera can only be monitored by local monitor with fixed monitor interface, a set of smart camera management system was developed independently, which was a combination of graphic programming function, task running function and data monitoring function. Based on the functional requirements of programming, running and monitoring of smart camera that were analyzed, structure of the system was designed. Graphic image processing unit and visual programming interface were built to realize graphic programming function while the data management model of system was also discussed. Finally, the feasibility of the system which was used on industrial manufacture was verified by using some examples. The results showed that the software reduces development time and lowers the cost of development, and could manage smart camera working on production line effectively.

Keywords: smart camera; software design; graphic programming; configuration

0 引言

随着计算机技术的发展和工业自动化水平的提高, 自动检测技术得到越来越广泛的重视和应用, 智能相机的应用也越来越广泛^[1-2]。作为自动化装备的“眼睛”, 智能相机应在不同的应用场景下定义不同的图像处理应用, 以满足不同的需求。若采用传统的文本编程方式, 开发人员需要深入了解图像处理算法接口, 掌握相关图像处理的知识, 这势必会增加编程难度和出错率^[3]。此外, 智能相机资源的管理、检测结果和状态的监视, 都是智能相机所必备的功能。在传统的本地监控方式中, 监控设备与智能相机以电缆连接, 这迫使工作人员难以实现远程操作和监视, 人力成本较高, 且监控界面固定, 难以定制。

针对当前的现状和市场需求, 本文采用面向对象的编程思想, 以 Microsoft Visual Studio 2012 为开发平台, 设计并实现

了一套集图形化编程、任务运行、远程监控于一身的智能相机管理系统。该系统以图形化编程方式实现了对智能相机的图像处理任务的编程定义、对监视设备界面布局的自由组态、对监控界面数据的自定义, 实现了编程过程中图像处理仿真结果的实时反馈, 实现了图像处理任务在运行环境中的准确执行以及远程设备对智能相机的监视和操控。图形化编程、编程与仿真并行的编程方式、远程监控的监控方式, 都节约了相机开发管理的人力物力成本, 有利于提高生产效率及其自动化程度^[4]。

1 系统功能与结构分析

1.1 系统功能分析

智能相机管理系统设计前, 需要对智能相机使用管理过程进行分析, 总结出其相关的功能需求, 其功能需求如下:

1) 图形化的相机开发工具。智能相机处理算法多种多样, 通过算法的组合与参数调整可以生成不同的应用实例。由于算法种类繁多, 若以传统的文本编程模式进行, 势必增加二次开发者的负担, 导致开发效率下降和出错率增加。因此, 开发工具应拥有简单友好的图形化开发环境, 提供算法模块的图形化编程模式和可视化的参数设定、信息录入功能。

2) 相机监视。工业智能相机应用于生产中的检测识别环节, 对于应用场景, 需要建立不同的检测识别应用; 针对不同的应用, 监视设备监视数据对象、来源也应做出不同的调整; 对于同一应用而言, 不同的监视级别对数据的需求也有差异。智能相机需提供可定制的数据监视模式, 由用户根据应用场景

收稿日期: 2017-05-09; 修回日期: 2017-05-23。

基金项目: 国家科技支撑计划资助项目(2015BAF20B01); 广东省科技计划资助项目(2011B010300002, 2016A010102008, 2015B010101005, 2014B090921003); 广州市科技计划资助项目(201508030007, 201604010064)。

作者简介: 吴培浩(1993-), 男, 广东潮州人, 硕士研究生, 主要从事嵌入式系统、计算机应用方向的研究。

李迪(1965-), 女, 山东青岛人, 教授, 博士生导师, 主要从事嵌入式系统、自动控制 and 机器视觉方向的研究。

定制监视数据, 满足不同应用环境的监控需求。

3) 仿真环境与独立运行环境。仿真环境在编程过程中实时反馈结果。帮助用户调整编程参数以获得合适的图像处理任务。而运行环境内嵌于智能相机中, 在用户启动智能相机后实现对相机资源的管理、图像处理任务的执行。用户完成编程工作后, 通过网络通讯或文件交互的方式实现编程工具与运行环境的数据交互, 实现智能相机应用程序的导入。

1.2 结构设计

根据智能相机管理系统的需求分析和应用情况, 将总体结构按照功能划分为设备层、配置层、监控层三部分, 三者关系和交互如图 1 所示。

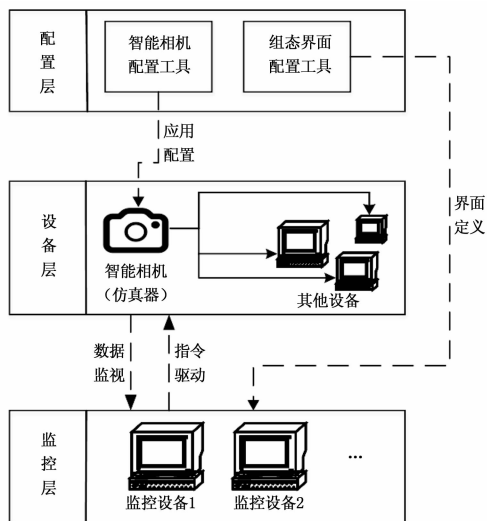


图 1 系统结构组成

1) 设备层: 主要由安装在生产线上或机械臂上的智能相机与其他外部设备如机器人等组成。作为系统的核心执行设备, 智能相机负责捕捉生产线上图像信息, 根据用户定义的处理任务对所得图像进行处理, 捕获图片特征信息, 并将处理结果传输至其他外部设备。

2) 配置层: 包括运行于 PC 机上的图形化相机编程工具和人机交互界面 (Human Machine Interface, 简称 HMI) 组态工具。配置层负责对设备层、监控层相关信息进行定义。智能相机编程工具主要定义智能相机的硬件参数、图像采集模块, 设计图像处理应用、设定算法模块参数、定义设备对外通信等; HMI 组态工具负责对监控界面进行组态, 定义监视数据, 设计界面消息响应以及控制指令输出。

3) 监控层: 主要为工控机、触摸屏或其他监视设备。HMI 组态工具针对不同的图像处理任务创建相应的监控界面, 配置结果由监视器解析运行, 通过预定义的界面元素和数据接口, 对设备层进行数据监视和指令驱动。

2 关键技术与实现

2.1 图形化软件设计

图形化编程软件是用户编辑图像处理应用和监控设备 HMI 的平台, 其设计主要包括智能相机配置工具和 HMI 组态工具的设计。本软件采用图形化编程思想, 旨在为用户提供一个友好的编辑环境, 方便用户进行应用开发。下文以智能相机

配置工具为例, 进行设计方案分析。

2.1.1 图元设计与管理

智能相机配置工具完成相机参数、图像采集、图像处理、数据通讯等方面的设置。图像处理应用的编辑是配置工具一大主要功能。为了方便用户使用, 采用图形化的编程方式。图像处理应用可由多个图像处理任务组合而成, 图像处理任务由不同的处理流程叠加而成。传统的文本编辑方式是直接调用图像处理函数进行设置, 而图形化编程方式则是将调用图像处理函数部分封装起来, 以图元和图形化界面向用户开放, 满足用户需求。

图元是一种具有输入和输出的可视化组件。在智能相机配置工具中, 图像处理任务以树形呈现, 而树节点本身是一种图元对象, 集合了图像、坐标与其他数据的输入输出端口^[5], 如图 2 所示。

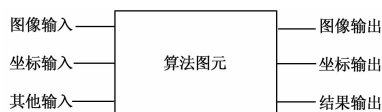


图 2 图像处理流程图元

设计时, 将图像处理算法抽象封装成图元对象, 对内管理数据结构、处理输入数据, 对外经由输出端口输出处理结果。而逻辑节点被抽象成控制图元, 形成树形分叉, 使得树形结构的组合更为自由多变, 满足应用需要。CImgTool 类是图元结构的父类, 其继承于 MFC 提供的 COBJECT 类, 包含了所有工具的通用数据结构, 并为所有算法工具定义了抽象的应用接口。图像处理工具直接或间接派生于 CImgTool 类, 对父类中定义的抽象接口如 WriteMemory () 进行了具体实现, 将特有的属性数据封装于其内部, 并将参数的设置、数据流动设置等以界面输入的形式呈现。

图元对象统一由 CImgTool * 指针统一管理调度, 利用 MFC 提供的 CTypedPtrArray 模板类进行图元对象指针的管理。使用时, 充分发挥了 C++ 面向对象编程的多态性, 利用基类指针 CImgTool * 调用抽象接口函数, 动态实现对相应具体函数接口的调用。

图元对象的配置是在上位机中进行, 而图像处理应用的运行是在设备中实现, 设备内部系统与上位机有一套一一对应的隐含图元对象, 保存着图元对象的相关数据。配置相关接口的实现在上位机中进行, 而图像处理接口的具体实现是在相机内部进行, 并通过通讯方式将结果同步到上位机图元对象内部数据中。其数据流如图 3 所示。

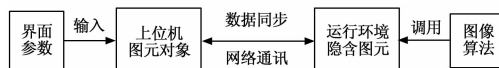


图 3 配置过程数据流动

2.1.2 界面与配置

图形化编辑环境的界面开发是在 Microsoft Visual Studio 2012 平台上进行, 利用 MFC 库提供的各类函数与框架进行界面设计, 简化设计工作, 提高设计效率。

软件的主界面设计采用 MFC 提供的 CFrameWndEx 框架类进行搭建, 除了提供常见的工具栏操作外, 界面主编辑区采

用界面分割技术, 将主界面分割成工具箱、应用编辑区和参数配置显示区, 在编辑时避免了弹窗式出现的多窗口层叠问题, 使界面更为简洁美观。

工具箱采用抽屉式布局, 利用鼠标点击工具抽屉的方式进行工具箱启闭操作。打开工具抽屉时, 抽屉内工具以列表形式呈现, 通过拖放至应用编辑区, 实现图像处理流程的创建^[6]。为了避开弹窗式的层叠界面, 右侧参数配置显示区采用界面复用方式实现。当选定任务树的某一节点, 右侧界面根据节点信息动态创建相应的数据界面, 以供用户配置。参数配置区由 CDispPaneContainer 类管理, 设置导航栏。根据用户选中对象, 由导航栏生成字符串 ID, 采用简单工厂模式对界面进行实例化, 从而完成界面的切换, 实现配置区域的复用。

2.2 通讯设计

通讯模块在智能相机管理系统的配置、管理中占据着举足轻重的作用, 它是确保数据传输的可靠性、实时性、稳定性的重要环节。本系统主要涉及的通讯包括智能相机与配置工具通讯、与监控层之间的通讯、与其他外部设备的通讯。

2.2.1 与配置工具通讯

在智能相机管理系统设计过程中, 智能相机的运行功能和配置功能被分开, 使两者在设计中更为独立合理。

配置工具是对相机执行功能进行设置定义的部分, 只完成相机配置工作。配置结果的有效性和准确性由智能相机(或仿真器)实时处理和反馈进行验证。两者之间通过以太网连接, 采用 TCP/IP 网络协议进行数据实时传输: 用户使用配置工具进行编程前, 首先需连接智能相机(或仿真器)。编程时, 配置工具将编程结果同步下载至智能相机(或仿真器), 智能相机(或仿真器)根据用户编程结果, 执行任务并将处理后的数据传输至配置工具并显示。用户可根据反馈结果, 调整参数, 以满足工业实际应用的需求。

由于考虑到通讯的实时性和独立性要求, 智能相机内部开辟一个用于与配置工具通讯的专用线程, 在配置工具中指定通讯智能相机的网络 IP 地址即可实现连接。同时, 通讯过程采用非阻塞的通讯方式, 设置超时限制, 以维护系统的稳定性。

2.2.2 与监控层通讯

智能相机作为监视设备的监视对象, 是监视设备的数据来源和交互对象。监视过程采用远程监视和操控的方式进行。监视设备通过网络通讯向智能相机发起连接, 发出数据请求指令和驱动指令, 并获取和显示由相机返回的监视数据, 从而实现了对相机的监视与操控, 同时, 基于网络通讯的监控方式也实现了监控设备与智能相机物理连接上的分离。

2.2.3 与其他设备通讯

智能相机相当于“眼睛”, 主要功能为获取图像数据信息, 并通过处理图像形成结论并指使其他设备完成工作, 如利用智能相机拍照和识别生产线上的物品, 并驱动机械手完成对物体的抓取和分类。因此, 与其他设备的通讯也是智能相机一大必备功能。

为了实现这一功能, 智能相机管理软件与硬件协调设计, 提供网络通讯、I/O 口的输入输出、EtherCAT 高速通讯协议、串口通讯、USB 通讯等多种通讯方式。智能相机管理软件将通讯相关的接口封装成与其他处理工具相似的通讯图元。编程时, 拖放通讯图元到任务树中, 从界面中设置通讯内容、通讯

格式和目标机信息, 即可完成对相机通讯模块设置。运行时, 由相机根据通讯类型和通讯信息, 实现与其他设备的通讯。

2.3 数据模型

运行环境的数据模型定义了数据的组织管理方式、数据流动形式, 主要涉及 3 个方面: 算法模块数据、节点间数据流动、监控数据流。具体实现如下:

1) 模块化算法工具。在图形化配置工具中, 各种算法抽象封装成图元, 将算法模块的相关参数和中间数据隐藏在图元内部, 提供数据修改接口, 与界面输入对接, 完成参数配置。运行环境为了与配置工具对接, 也采用面向对象的管理方案, 将算法分散并封装成不同的类。图像处理应用分若干任务模块, 每个任务模块由若干算法模块组成。任务的执行被拆分成若干子流程, 分别由不同算法模块负责执行, 实现高内聚低耦合的特点, 提高了算法数据的安全性、算法功能的复用性。

2) 节点间数据流。对于图像处理任务而言, 任务中各算法模块之间存在许多联系, 前驱工具产生的数据通常作为后驱工具的输入, 包括图像、坐标系或某些其他数据。算法工具输入输出数据结构在类中的定义如下所示:

```
class CImgTool: public CObject
{
protected:
    const CImage *    m_pImgInput; // 图像输入
    const UserCoordinates *    m_pCoordSys; // 坐标输入端口
    std::list<CVarPort> m_VarList; // 数据输入
    std::list<CLinkPort * >    m_ImgOutPortList; // 图像输出列表
    std::list<CLinkPort * >    m_OrgOutPortList; // 坐标输出列表
    std::list<CLinkPort * >    m_LogOutPortList; // 逻辑输出列表
    std::list<CLinkPort * >    m_VarOutPortList; // 一般的数据变量输出列表
}
```

算法工具中封装着数据端口的输入端和输出端, 构建任务时, 由系统自动为其搭建连接, 形成完整的数据流, 保证算法工具数据完整和正常运行。

3) 监控数据流。运行环境是与监视器通讯过程中的服务端。通讯过程分两部分: 当监控设备处于配置阶段时, 运行环境通过调用函数 GetMonitorList() 获取数据通道并通知 HMI 组态工具。当监控设备处于运行阶段, 运行环境根据数据请求指令, 调用 GetMonitorData() 函数获取监控数据, 以供监控设备监视。

3 应用

使用本文研制的智能相机管理系统, 以分拣系统为例, 某生产线上利用智能相机, 对传送带上工件颜色进行检测, 对指定颜色产品进行分类并利用机械臂进行抓取和分类, 同时统计各类产品的数量。针对这一任务, 设计图像处理任务和监控界面: 捕捉工件图片, 对输入图片做滤波处理; 查找指定颜色的工件, 若查找成功, 则将工件的中心坐标转化为机械手空间坐标后传输至机械手, 由机械手完成抓取分类工作, 并统计个数; 若查找失败, 即无目标工件, 则放弃抓取。编程结果如图 4 所示。

编程完成后, 将程序导入智能相机中, 启动监控设备, 利用监控设备驱动智能相机运行, 并监视其运行结果, 监控界面如图 5 所示。

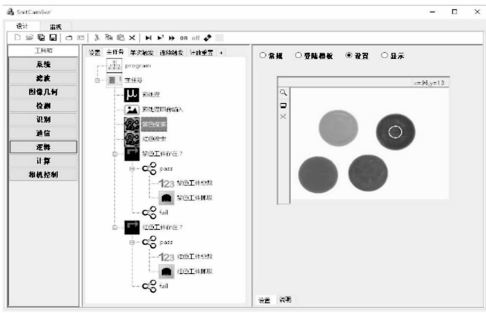


图 4 工件分拣编程界面

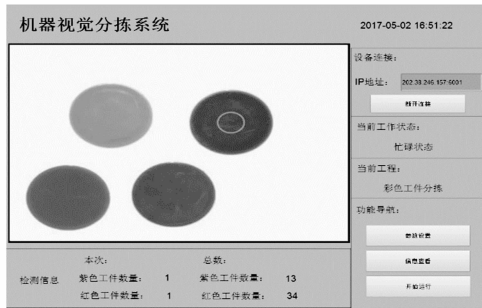


图 5 工件分拣监控界面

(上接第 179 页)

量与经济的快速增长, 并通过企业技术进步保持企业碳排放强度的合理下降, 则企业碳排放峰值出现的时间在 2030 年~2035 年之间。如果企业在经济发展时企业碳排放强度降低, 会延后企业碳排放峰值出现时间。

从表 3 可以看出灰色神经网络可以全面考虑影响企业碳排放峰值的因素, 并且准确地预测出企业碳排放峰值出现的时间与峰值额。仿真实验证明, 本文所提模型可以准确预测企业碳排放峰值, 并有效帮助企业设计碳排放减排路径。

表 3 灰色神经网络模型在各种模式下峰值预测结果

情景模式	峰值出现的年份/年	峰值额/万吨	碳排放量强度/万吨
低等基准变化速率模式	2020	523 154.22	86 497.3
中等基准变化速率模式	2035 年内没有出现峰值	—	—
高等基准变化速率模式	2035 年内没有出现峰值	—	—
节能情景模式	2035 年内没有出现峰值	—	—
失控情景模式	2035 年内没有出现峰值	—	—
环保情景模式	2030	376 856.34	46 537.67
低碳情景模式 A	2025	465 745.23	105 647.45
低碳情景模式 B	2035 年内没有出现峰值	—	—

3 结束语

采用当前的预测模型 TFDI 模型对企业碳排放峰值进行预

4 结语

本文基于图形化编程思想与网络通讯方式, 开发了一套智能相机管理系统。该系统集相机图形化编程、相机运行管理、相机监控等功能于一身, 通过拖放式的编程方式进行智能相机编程, 通过网络通讯方式实现数据同步、远程监控等, 为用户提供一套操作简单便捷、功能完整的智能相机编程管理系统。经实验验证, 该系统在生产应用中具有良好的实用价值, 值得进一步推广应用。

参考文献:

- [1] 裴志军, 曹继华, 张骥祥. 嵌入式机器视觉中的智能相机技术 [J]. 天津工程师范学院学报, 2008, 18 (2): 5-8.
- [2] 刘 伟. 一种基于智能相机的机器视觉系统的设计 [J]. 重庆工商大学学报: 自然科学版, 2013, 30 (11): 66-69.
- [3] 吕 敏, 沙 莎. 可视化编程数字图像处理平台的界面设计与实现 [J]. 计算机系统应用, 2010, 19 (9).
- [4] 王腾腾. 基于图编程的嵌入式机器视觉算法平台研究 [D]. 杭州: 杭州电子科技大学, 2016.
- [5] 杨林蛟, 张 鑫. 基于 JGraph 的图像处理编程平台的实现 [J]. 机电工程, 2011, 28 (8): 978-982.
- [6] 杨林蛟, 张 鑫. 基于组件的医学影像处理算法平台的实现 [J]. 微处理机, 2010 (2): 97-100, 104.

测, 由于预测过程中无法全面考虑企业碳的排放影响因素, 容易使企业碳排放的预测出现误差。提出一种基于灰色神经网络模型的企业碳排放峰值预测模型。并通过仿真实验证明, 所提模型可以准确地预测企业碳排放的峰值, 有效地帮助企业设计碳排放减排路径。

参考文献:

- [1] 朱宇恩, 李丽芬, 贺思思, 等. 基于 IPAT 模型和情景分析法的山西省碳排放峰值年预测 [J]. 资源科学, 2016, 38 (12): 2316-2325.
- [2] 陈 盈, 李函昱, 田立新. 江苏省能源消费结构对碳排放影响的效应与趋势——基于 LMDI 和 Logistics 模型的实证研究 [J]. 数学的实践与认识, 2016, 46 (21): 158-170.
- [3] 王秋平, 舒 勤, 黄宏光, 等. 含有误差校正的小波神经网络交通流量预测 [J]. 计算机测量与控制, 2016, 24 (2): 168-170.
- [4] 侯连浪, 梁利喜, 刘向君, 等. 基于 BP 神经网络的页岩静弹性模量预测研究 [J]. 科学技术与工程, 2016, 16 (30): 176-180.
- [5] 贺永俊, 杨 珊. 基于 QPSO 灰色神经网络的基坑锚固荷载预测 [J]. 科技通报, 2016, 32 (1): 109-113.
- [6] 刘晓辉, 闫二旺. 能源与产业结构调整下我国工业碳排放峰值调节机制研究 [J]. 工业技术经济, 2016, 35 (12): 107-114.
- [7] 周晨吕, 胡 静, 李立峰. 崇明岛中长期碳排放预测及其影响因素分析 [J]. 长江流域资源与环境, 2015, 24 (4): 632-639.
- [8] 朱松丽. 基于国土空间开发格局的城市化地区和其他功能区碳排放现状——黔粤案例研究 [J]. 气候变化研究进展, 2016, 12 (2): 132-138.
- [9] 郭 斌, 许江宁, 何泓洋, 等. 基于微分器与 RBF 神经网络补偿的 PID 控制 [J]. 计算机仿真, 2016, 33 (6): 294-297.
- [10] 朱卫平, 汪志成, 袁晓冬, 等. 一种光伏功率预测的 RBF 神经网络校正模型 [J]. 电子设计工程, 2016, 24 (13): 113-115.