

基于嵌入式图像处理系统的水产养殖轨迹跟踪

许敏界, 叶宏宝, 徐志福, 华珊*

浙江省农业科学院农业装备研究所, 浙江 杭州

收稿日期: 2022年9月20日; 录用日期: 2022年10月1日; 发布日期: 2022年10月24日

摘要

目前, 在传统水产养殖向产业化、集约化发展的过程中, 监测技术与方法是薄弱环节, 因此, 研究和应用水产养殖轨迹跟踪技术具有重要意义。本文主要研究了基于嵌入式图像处理系统的水产养殖轨迹跟踪技术, 提出了一种水产养殖在线监测系统的设计方案。该系统基于嵌入式技术, 结合GSM/GPRS通信和射频无线传输技术, 主要实现对水生鱼类的实时轨迹跟踪和监控。智能化设计使监控人员能够根据异常数据及时采取有效措施。本文的实验数据表明, 在X轴速度为300 mm/min的条件下, 水产养殖轨迹跟踪系统的跟踪误差达到0.5 mm或更小, 满足设计要求。该系统结构简单, 功能易于扩展, 测量数据误差小, 适用于水产养殖产业化领域。

关键词

图像处理, 轨迹跟踪, 嵌入式系统, 鱼类跟踪, 运动目标检测

Trajectory Tracking of Aquatic Fish Breeding Based on Embedded Image Processing System

Minjie Xu, Hongbao Ye, Zhifu Xu, Shan Hua*

Institute of Agricultural Equipment, Zhejiang Academy of Agricultural Sciences, Hangzhou Zhejiang

Received: Sep. 20th, 2022; accepted: Oct. 1st, 2022; published: Oct. 24th, 2022

Abstract

At present, the monitoring techniques and methods are weak in the process of traditional aquatic fish farming towards industrialization and intensive development, so the research and implementation of aquatic fish farming trajectory tracking technology are of great significance. This

*通讯作者。

paper mainly studies the tracking of aquatic fish farming based on an embedded image processing system and proposes a design scheme for an aquaculture online monitoring system. This system is based on embedded technology, combined with GSM/GPRS communication and radio frequency wireless transmission technology, and mainly realizes real-time trajectory tracking and monitoring of aquatic fish. The intelligent design enables monitoring personnel to take effective measures in time according to abnormal data. The experimental data of this paper shows that the tracking error of the aquatic fish breeding trajectory tracking system reached 0.5 mm or less under the condition of an X-axis speed of 300 mm/min, which meets the design requirements. The experimental results in this paper show that the system is simple in structure, easy to expand in function, have small errors in the measured data and, is suitable for the field of industrialized aquatic fish breeding.

Keywords

Image Processing, Trajectory Tracking, Embedded System, Fish Tracking, Moving Target Detection

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0)

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着时代的发展,我国水产养殖品种不断创新,产量不断增加。水产养殖生产自动化管理技术的研究与应用也越来越受到重视。保持养殖场水环境的稳定,即监测和控制养殖环境中的溶解氧、水温和 pH 值等水质指标,对水生生物的生长和水产养殖业的发展具有重要意义。

近年来,图像处理、无线通信技术和嵌入式技术在水产养殖生产和管理控制中的广泛应用,增强了鱼类养殖的自动监控能力,提高了水产养殖行业的生产效率和经济效益[1] [2]。水产养殖智能监测系统通常需要完成数据的采集和传输,并提供方便查询的接口,还需要满足低成本、低能耗的要求[3] [4],并利用图像采集设备与图像处理方法实现智能分析的功能[5] [6] [7]。本文设计了一种基于嵌入式图像处理系统的水产养殖轨迹跟踪系统,实现了水产养殖轨迹的自动跟踪。该系统采用 USB 摄像头作为视觉传感器提取鱼的轨迹图像,并将图像输入嵌入式微处理器进行处理,获取鱼的轨迹信息。通过轨迹信息提取跟踪轨迹与轨迹之间的位置偏差,采用相应的控制算法计算输出的控制量,并调整跟踪轨迹参数的大小,达到轨迹跟踪的目的。

2. 图像处理技术

2.1. 运动目标检测算法

1) 背景差分法

通过将目标图像与背景图像相减来检测目标,即从背景图像中提取图像中的变化区域。背景差分法的关键问题是建立背景模型,特别适用于静态场景中的运动目标检测。其基本思想是假设当前帧中的运动对象图像为 $f_k(x, y)$, 获取的背景图像为 $B_k(x, y)$, 并将两帧相减以获得差分结果 $D_k(x, y)$ 。

$$D_k(x, y) = |f_k(x, y) - B_k(x, y)| \quad (1)$$

由于图像受各种噪声的影响,背景相减的结果不能全部为 0,为了减小噪声的影响,需要对 $D_k(x, y)$ 进行阈值化处理。如果 $D_k(x, y)$ 大于阈值,则可以判断该点是运动目标区域。

$$D_k(x, y) = \begin{cases} 1 & D_k(x, y) > TH \\ 0 & \text{else} \end{cases} \quad (2)$$

背景差分法成功检测到运动目标区域，需要注意以下几点：

a) 背景模型可以通过使用包含运动对象的图像构造背景模型来获得，例如，使用多帧图像的平均值来找到背景模型。

b) 由于噪声的影响，我们在获取视频图像的过程中通常会引入噪声。光和风等环境因素的变化将导致背景发生变化。因此，我们可以根据噪声源采用合适的滤波器或自适应动态背景更新方法来降低噪声的影响。

c) 在阈值的选择上，只有选择合适的阈值，才能准确地检测出运动目标，而阈值的选择一般是根据经验来选择的。当存在外部干扰时，背景模型会发生变化，因此需要实时更新背景模型。背景模型的建立通常采用平均背景模型和统计背景模型。

2) 帧间差分法

帧间差分法利用帧间的差分信息，其中信息是视频序列图像中变化很大的像素。如果记录了这些像素，则可以检测到运动目标区域[8]。因此，帧间差分法是基于变化检测运动目标。首先，比较序列图像的两帧或更多帧，然后比较差分结果的像素差分，并设置适当的阈值以提取运动目标[9]。让二值差 $D_f(x, y)$ 表示第 k 帧的图像 $f_k(x, y)$ 和第 $k+1$ 帧的图像 $f_{k+1}(x, y)$ 的变化：

$$D_f(x, y) = \begin{cases} 1 & |f_{k+1}(x, y) - f_k(x, y)| > T \\ 0 & \text{other} \end{cases} \quad (3)$$

在等式(3)中， (x, y) 表示像素位置， T 表示差分结果的阈值。当为0时，表示图像前后两帧的亮度值没有变化；当为1时，表示图像前后两帧的变化区域由于运动而发生变化。

帧间差分法对场景中光照的变化具有良好的抗干扰性，不需要实时建立和更新背景，适应性强，算法简单，检测速度快，实时性高。但当目标运动速度较快或较慢时，会出现过分割或欠分割现象；其次，当目标灰度分布均匀时，目标重叠区域会产生较大的空洞。帧差法只能检测运动目标的近似位置，且检测到的目标不完整[10]。

3) 光流法

光流场定义为序列图像中目标运动产生的瞬时位移场。光流是指图像中亮度运动的分布。当物体离开图像时，图像中光流速度的变化是连续的；当物体与图像背景发生相对位移运动时，物体与背景形成的速度矢量不同，可以检测物体的失准和背景差。然而，用它来检测运动目标需要较长的时间，实用性和实时性较差。光流包含目标的运动信息和场景的三维结构信息，因此基于光流的分割方法通过计算三维运动场映射到二维图像，即，根据与场景中物体运动的关系，判断图像中像素的灰度值随时间变化，计算运动目标的光流矢量以检测运动目标[11] [12]。

假设在时间 t 时图像像素 (x, y) 的灰度值为 $f(x, y, t)$ ，并且在时间 $t+dt$ 时 (x, y) 的位置为 $(x+dx, y+dy)$ ，当 dt 小时，可以假设两点的灰度值不变，即：

$$f(x, y, t) = f(x+dx, y+dy, t+dt) \quad (4)$$

用泰勒级数展开方程(4)，忽略高阶项以求极限：

$$f(x, y, t) = \frac{\partial f}{\partial x}u + \frac{\partial f}{\partial y}v + \frac{\partial f}{\partial t} + f(x, y, t) \quad (5)$$

在等式(5)中， u 和 v 分别是点在 X 和 Y 方向上的移动速度。根据约束条件，光流计算方法大致可分为三类：1) 基于匹配的光流计算方法，该算法一般采用面积和特征两种方法进行匹配，该方法需要对目

标的主要特征进行连续的定位和跟踪,该方法能够检测出较大的目标和亮度变化较大的目标。2) 基于频域中的光流方法,相位的存在或不存在可以通过滤波器组的输出相位或频率来确定,滤波器组的速度可以调整。它涉及复杂和繁琐的计算,并且不容易进行可靠性评估。3) 基于梯度的光流方法。该方法通过对图像进行区域分异来计算二维速度场。由于其算法相对简单,对运动目标的检测效果较好,因此得到了广泛的研究。光流方法可以精确地划分运动区域,但大多数光流方法计算量大,抗噪性和实时性差。

2.2. 运动目标检测的难点

在视频序列图像中检测运动目标时,影响检测结果的因素很多,使得其检测技术复杂而特殊。在实际生活中,运动物体的检测主要存在3类问题:

1) 由于光线问题,物体在运动过程中产生阴影干扰,干扰部分与物体的目标运动一致,二者与背景差异很大,因此,阴影部分会被归类为运动目标。

2) 动态背景的干扰。在视频序列图像中,除了运动外,目标不是静止的。例如,在室外监控中,树来回摆动。行为目标的提取极其困难,最终导致监测系统失效。

3) 场景照明、天气、照明的变化以及摄影机光圈的调整都会影响照明的变化,从而导致图像的变化,使提取的运动目标不准确。

针对以上三个难点,本文认为基于颜色特征的方法可以解决第一个问题,即在颜色特征空间中使用多个阈值来检测前景、背景和阴影;背景差分法可以解决第二个问题,即背景模型实时更新,背景变化时模型也会发生变化;帧间差分法可以解决第三个问题。从原理上可以看出,帧间差分法对场景中的光照变化具有较好的抗干扰能力。

2.3. 嵌入式操作系统

目前,流行的嵌入式操作系统主要有 Windows CE, VxWorks, uC/OS-II 和 Linux。开发人员通常根据实际情况选择不同的开发系统。本文根据实际养殖需求,对各个操作系统进行了全面的比较,最后选择嵌入式 Linux 作为开发平台的操作系统。嵌入式 Linux 是标准 Linux 在嵌入式平台上的移植。它继承了标准 Linux 的优点,随着嵌入式技术的发展,近年来成为一个应用热点。嵌入式 Linux 具有以下主要功能:

1) 支持多个 CPU。嵌入式 Linux 得到了广泛的应用,支持目前市场上几乎所有的嵌入式微处理器。如 ARM、BMPROERPC、摩托罗拉等。

2) 高性能、可定制的内核。内核可以根据特定的硬件进行定制,这在运行时需要更少的资源,并且稳定高效。独特的模块化机制可以动态地将用户设计的模块插入内核,并可以处理复杂的多任务需求。

3) 可以提供图形界面工具。支持高端图形适配器和显示器。

4) 它附带了大量常用的硬件设备驱动模块,如 LCD、USB 摄像头等,以方便用户应用程序开发。

5) 开源代码,广泛支持。只要任何人遵守 GNU 通用许可证(GPL),他们就可以免费获得和使用 Linux 内核,大大降低了开发成本。由于其开源特性,其用户群最大,更容易找到相关信息,这为应用程序开发带来了便利。目前市场上流行的嵌入式 Linux 版本主要有 ReadHat Linux、Debian Linu、FedoraLinux 等。其中,红帽 Linux 是中国用户最多的版本。与其他版本相比,它的信息更完整,更易于开发,性能稳定。本文最终选择 Redhat 9.0 作为此次开发的嵌入式操作平台。

3. 实验

3.1. 实验设计

基于 Matlab 软件平台,开发用于跟踪鱼类的运动轨迹,输出水生鱼类的运动行为参数的应用程序,可以实时、准确地输出水生鱼类的运动轨迹、运动速度和加速度,并进行格式化和保存。基于 Matlab 的

视频图像跟踪方法的基本流程包括视频输入、图像处理、鱼的轨迹、速度和加速度的数据输出。该方法将垂直狭缝鱼游动的视频导入 Matlab 程序,通过对视频图像的灰度、混合高斯建模和背景差分进行处理,采用块自适应阈值获得较好的图像;并对处理后的图像进行连通域分析和质心点确定,最终输出水生鱼类的运动轨迹、运动速度、加速度等数据。

1) 读取视频图像的每一帧,将图像转换为灰度空间,得到灰度图像,使用混合高斯背景模型对灰度图像进行背景建模,得到背景图像。

2) 采用背景差分法提取前景图像,以垂直狭缝鱼道作为感兴趣区域(ROI),仅提取 ROI 中的运动前景;对 ROI 中的运动前景进行区域滤波,去除小噪声干扰。

3) 由于垂直狭缝鱼道中存在湍流和阴影现象,鱼道中目标鱼的比例较小,因此固定的全局阈值分割方法无法达到理想的分割效果。本文采用块自适应阈值法(一种针对不同图像区域选择不同分割阈值的方法)对图像进行分割。该方法抗噪声能力强,对垂直缝鱼道分割效果好。

4) 对处理后的图像进行形态腐蚀扩展操作和连通域分析,将最大连通区域的轮廓作为单个鱼区域,找到该区域的最小外接矩形,并将其作为最终鱼体区域,以质心点所在的区域作为单个鱼的位置;可以对图像的每一帧执行相同的跟踪操作,以获得鱼的轨迹。

5) 手动计数是指通过建立直角坐标系将视频导入 Logger Pro 视频分析软件,然后逐帧手动点击视频中的单个鱼,获取鱼在不同时刻的运动位置,手动获取目标鱼在所有时刻的运动,即鱼在鱼道中的轨迹形成等运动参数。

3.2. 跟踪图像采集

图像采集模块的主要功能是通过视觉传感器获取轨迹信息,为后续的轨迹跟踪做好准备。在实际应用中,可以根据实际需要调整相应的参数,以获得最佳的现场信息。本文选用的视觉传感器是基于 USB 接口的 CMOS 视觉传感器。主芯片采用中兴微生产的 ZC0301P 系列。Linux 内核中提供了对它的驱动程序支持,只需对内核进行简单配置,即可使嵌入式开发系统支持 USB 摄像头。Linux 系统提供专门为图像采集提供的标准接口 V4L2。在设计图像采集程序时,可以基于 V4L2 的 API 进行开发。

3.3. 弹道图像处理

为了实现轨迹跟踪,如何对采集到的轨迹图像进行处理以获取位置信息是一个极其关键的步骤。下面将详细介绍如何实现图像信息的处理。

1) JPEG 格式图像解码

为了提高图像采集和传输的效率,在图像采集设置中,本文将采集的图像格式设置为 jpeg,后续处理要求图像为 bmp (RGB)格式。在图像处理之前,需要对图像进行解码。本文选择 Libjpeg 库对 jpeg 格式的图像进行解码。在解码工作开始之前,需要建立相应的编译环境。需要将四个文件 jconfig.h、jmorecfg.h、libjpeg.lib、jpeglib.h 复制到项目文件夹中,并在程序头文件中添加 jpeglib.h。

2) 图像处理

基于视觉的轨迹跟踪系统的主要任务是实现对轨迹的实时、准确跟踪。弹道位置信息的快速提取与识别是整个系统的关键环节。研究如何在嵌入式开发平台下实现快速、准确的图像处理是非常重要的。本文选用的图像处理技术主要包括灰度图像、模糊去噪、阈值分割、轨迹特征提取等。

4. 讨论

4.1. 跟踪性能测试分析

本文研究的最终目的是实现对鱼类运动轨迹的精确跟踪。在图像反馈的基础上,如何设计合理的控

制算法和控制程序是本文研究的重点和难点。在本文中，比例参数在系统中的作用是提高响应速度，减小稳态误差，但会增加启动时的振荡；积分参数在系统中的作用是消除系统稳态误差，延迟系统稳定时间；差动参数的功能是抑制超调，但可能导致振荡和不平衡。合理的参数选择对控制系统的稳定性有着至关重要的影响。本文采用临界比例法对 PID 控制器的参数进行整定。首先，仅采用比例控制来调整 K_p 值，使系统出现临界振荡状态。此时， K_p 值记录为 K_r ，一次振荡时间记录为 T_r ，采样周期记录为 T_s 。可通过查看临界比法参数表进行参数设置，如表 1 所示。计算后，确定 PID 控制器三个参数的近似范围。经过多次实验和对计算参数的调整，最终确定该控制系统的三个参数为： $K_p = 2$ ， $K_i = 0.3$ ， $K_d = 0.5$ 。在跟踪水生鱼类过程中，X 轴(移动机器人小车的轮子)以固定速度移动。本文将 X 轴的移动速度设置为 300 mm/min，通过调整 Y 轴，实现轨迹的跟踪。在跟踪过程开始之前，它将被手动调整。通常，要跟踪的轨迹线在 USB 摄像头提取的图像中心的位置被选择为初始位置。在跟踪过程中，当 Y 轴上的距离偏离要跟踪的轨迹时，由于相机固定在 Y 轴上，此时提取的图像将显示一定的偏移。计算出的偏差由 PID 算法计算，最后转换成控制系统发送的脉冲信号，调整 Y 轴的运动，实现对轨迹的跟踪。

Table 1. PID parameter setting test data
表 1. PID 参数设置测试数据

控制器类型	K_p	K_i	K_d
P 控制器	$0.5 K_r$		
PI 控制器	$0.45 K_r$	$\frac{K_p T_s}{0.85 T_r}$	
PID 控制器	$0.6 K_r$	$\frac{K_p T_s}{0.5 T_r}$	$\frac{0.125 K_p T_d}{T_s}$

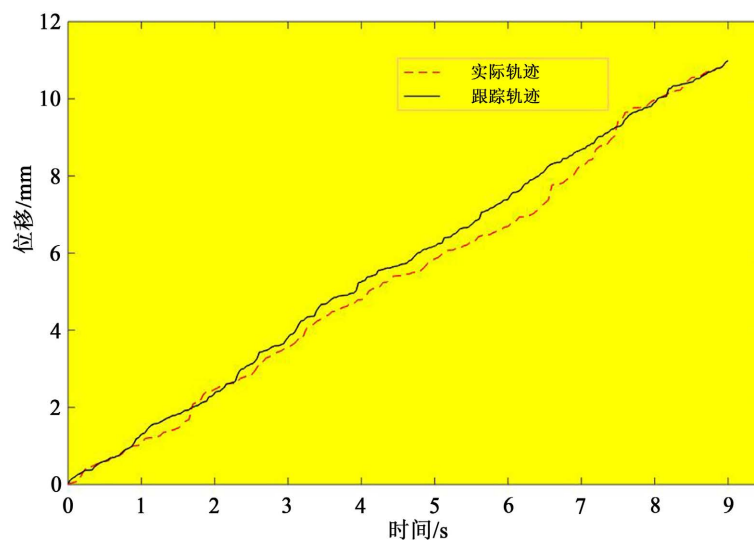


Figure 1. Trajectory tracking effect
图 1. 轨迹跟踪效果

如图 1 所示，是系统启动后从开始跟踪到稳定运行的跟踪效果图。在图中，X 轴表示时间，Y 轴表示位移。从图中可以看出，当 X 轴速度为 300 mm/min 时，跟踪系统能够更快地稳定，稳定后系统跟踪误差小于 0.5 mm。如图 2 所示，这是一个跟踪曲线图，分别显示了跟踪开始时的偏差曲线和速度曲线。

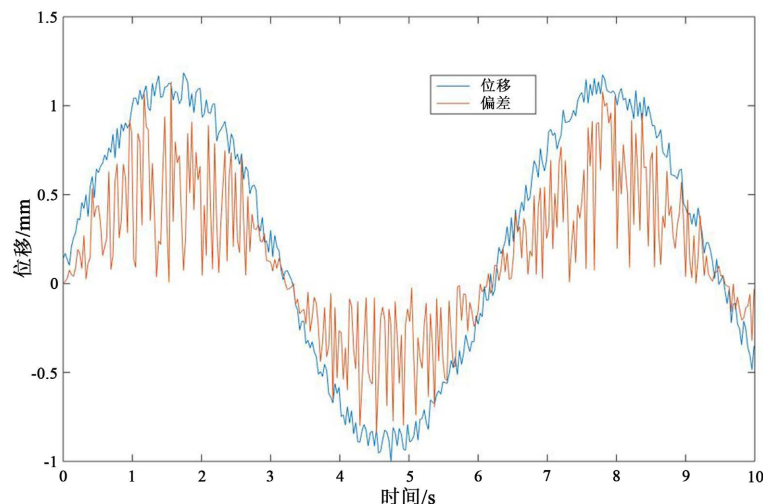


Figure 2. Trajectory tracking curve
图 2. 轨迹跟踪曲线

经过系统性能测试, 轨迹跟踪系统以 300 mm/min 的速度实现了对被跟踪轨迹的高精度跟踪, 但在跟踪阶段开始时, 跟踪性能较差。如图 2 所示, 存在较大偏差和冲击, 而速度的急剧变化将影响机械结构的使用寿命。为了提高系统的跟踪性能, 同时起到保护机械的作用, 有必要在后续研究中分析误差和振荡的原因, 并对控制算法进行适当调整, 以优化系统的跟踪性能。

经过系统性能测试, 轨迹跟踪系统以 300 mm/min 的速度实现了对被跟踪轨迹的高精度跟踪, 但在跟踪阶段开始时, 跟踪性能较差。

误差的主要原因为以下几点:

1) 图像采集与处理的滞后性

在移动机器人小车开始移动之前, 每个轴都会通过手动调整调整到初始位置, 通常初始位置是轨迹在采集图像中心的位置。当发送启动信号时, 移动机器人小车的 X 轴开始以固定速度前进。此时, 轨迹跟踪主要通过调整 Y 轴来实现。

在移动时间 $T = 0$ 时, 发送开始跟踪的信号, 车辆开始移动, 执行图像采集 1, 获取时间 0 时的轨迹信号, 并在采集后执行图像处理 1。在 $T = 200$ ms 时, 根据图像处理 1 的结果执行控制 1 和图像采集 2。由于图像处理 1 在时间 0 处理位置信息, 偏差为 0, 因此将根据偏差 0 控制运动控制 1。在 $T = 400$ ms 时, 根据图像处理 2 的结果执行控制, 并且开始图像获取 3。图像处理 2 在 200 ms 时处理位置信息, 并且存在偏差, 由运动控制 2 进行调整。在 $T = 600$ 时, 启动运动控制 3, 然后运动控制 2 完成, 调整是在 $T = 200$ ms 时产生的偏移。也就是说, $T = 200$ ms 处的偏差需要等待 $T = 600$ ms 才能完成调整。从以上分析可以看出, 在开始运行时, 由于图像采集和图像处理引起的延迟, 系统控制将有 400 ms 的滞后, 这将极大地影响系统在开始运行时的稳定性。

2) 由图像处理引起的错误

跟踪系统基于图像处理反馈的位置偏差信息进行控制。控制器的计算基于图像处理的结果, 图像处理反馈的信息被视为完全准确的信息。但在实际应用中, 图像处理存在一定的误差。特别是当图像中存在灰度值与待跟踪轨迹相似的干扰点时, 误差会更大。如图 3 所示, 直线拟合后, 图中直线斜率为 -0.0893 , 根据不同斜率计算的物理坐标位置偏差达到 0.1 mm。这部分误差主要影响系统达到稳定状态后的跟踪稳定性。



Figure 3. The influence of interference on image processing
图 3. 干扰对图像处理的影响

4.2. 系统稳定性分析

如表 2 所示, 水质监测设备非常稳定, 几乎没有奇异点。进一步检测了系统轨迹跟踪的精度和精度, 校准结果如表 3 所示。

Table 2. Fish pond test data results
表 2. 鱼塘试验数据结果

时间	水温(°C)	pH	溶解氧(mg/L)	氨氮(PPM)
1	25.11	7.98	7.66	0.17
2	25.62	7.88	7.56	0.17
3	25.03	7.86	7.63	0.18
4	25.13	7.85	7.54	0.18
5	25.14	7.71	7.43	0.19
6	25.25	7.75	7.37	0.19
7	24.78	7.76	7.32	0.18
8	24.25	7.76	7.23	0.19
9	24.78	7.74	7.25	0.20
10	24.99	7.64	7.21	0.21

Table 3. Statistics of trajectory tracking effect
表 3. 轨迹跟踪效果统计

实际轨迹效果	跟踪效果	跟踪误差(%)
20	19.41	2.95
22	21.39	2.81
24	23.36	2.67
26	25.30	2.69
28	27.19	2.91
30	29.17	3.10

数据错误可能会影响系统可靠性。当主机接收到数据时,可能会由于环境突变、设备故障或程序错误导致单个参数发生突变,远远超过数据的平均值,从而影响数据分析和处理。本文轨迹跟踪参数的误差在5%以内,能够满足实际要求,如表3所示。

5. 结论

本文研究的嵌入式轨迹跟踪系统在X轴速度为300 mm/min时,跟踪误差小于0.5 mm,满足设计要求。它具有体积小、成本低、维护使用方便等优点,具有广阔的市场前景。根据嵌入式系统自身硬件的特点,在研究图像处理算法时,系统选择了一种简单的算法。在干扰小、光线好的情况下,系统跟踪性能理想;当现场环境较差时,系统的跟踪效果往往不令人满意。在后续研究中,图像处理方法有待改进。

本文将远程多点轨迹跟踪技术应用于设施养鱼的轨迹跟踪中,设计了一套低成本、高性价比、抗干扰能力强、结构简单、易于扩展的多通道远程鱼类关键参数轨迹跟踪系统。实践表明,该系统实用性强,可以为水产养殖用户提供可靠的实时鱼类定位参数变化信息。根据系统采集的坐标参数信息,操作人员可以科学管理养殖池塘,合理控制鱼类群体,提供适宜的生长环境,保证鱼类高产优质,为企业带来一定的经济效益和良好的社会效益。

参考文献

- [1] Hu, X.F., Feng, G., Duan, S.K. and Liu, L. (2017) A Memristive Multilayer Cellular Neural Network with Applications to Image Processing. *IEEE Transactions on Neural Networks & Learning Systems*, **28**, 1889-1901. <https://doi.org/10.1109/TNNLS.2016.2552640>
- [2] Ilmini, K. and Fernando, T. (2016) Persons' Personality Traits Recognition Using Machine Learning Algorithms and Image Processing Techniques. *Advances in Computer Science*, **5**, 40-44.
- [3] Liu, T., Wu, W., Chen, W., Sun, C.M., Zhu, X.K. and Guo, W.S. (2016) Automated Image-Processing for Counting Seedlings in a Wheat Field. *Precision Agriculture*, **17**, 392-406. <https://doi.org/10.1007/s11119-015-9425-6>
- [4] Kamoshida, Y., Nyukai, J., Habuka, K. and Hiraga, M. (2017) Image Processing Device, Image Processing Method and Storage Medium. *Fluessiges Obst*, **2**, 1-18.
- [5] Aoki, K. (2018) Server for Implementing Image Processing Functions Requested by a Printing Device. *Environmental Pollution*, **152**, 543-552.
- [6] Mortari, D., D'Souza, C.N. and Zanetti, R. (2016) Image Processing of Illuminated Ellipsoid. *Journal of Spacecraft & Rockets*, **53**, 448-456. <https://doi.org/10.2514/1.A33342>
- [7] Pu, J., Bell, S., Yang, X., Setter, J., Richardson, S. and Ragan-Kelley, J. and Horowitz, M. (2016) Programming Heterogeneous Systems from an Image Processing Dsl. *Acm Transactions on Architecture & Code Optimization*, **14**, Article No. 26. <https://doi.org/10.1145/3107953>
- [8] Satoh, J., Yamagishi, K., Nakashima, K., Katsura, K., Miyamoto, T., Watabe, M. and Ohmura, K. (2000) Data Processing System and Image Processing System. European Patent No. 99124662.0.
- [9] Kumari, D. and Kaur, K. (2016) A Survey on Stereo Matching Techniques for 3D Vision in Image Processing. *International Journal of Engineering & Manufacturing*, **6**, 40-49. <https://doi.org/10.5815/ijem.2016.04.05>
- [10] Noguchi, T., Shibara, T., Toemura, T., Ogino, M. and Murashita, K. (2018) Ultrasonic Diagnostic Apparatus and Image Processing Method. Japanese Patent No. 2017520689.
- [11] Yamada, K. (2016) Image Processing Method, Image Processing Device, and Image Processing Program. *Fuji Xerox*, **63**, 189-197.
- [12] Green, N., Glatt, V., Tetsworth, K., Wilson, L.J. and Grant, C.A. (2016) A Practical Guide to Image Processing in the Creation of 3D Models for Orthopedics. *Techniques in Orthopaedics*, **31**, 153-163. <https://doi.org/10.1097/BTO.000000000000181>