

การพัฒนาซอฟต์แวร์ตรวจจับวัตถุเคลื่อนที่และบันทึกภาพอัจฉริยะด้วยกล้อง webcam เพื่อการประยุกต์ใช้ในความปลอดภัยขององค์กรธุรกิจ

วิทยา บุญสุข^{1*}, จิม ยีนนาน¹ และ วัฒนา ศรีวรมย์¹

Development of Intelligent Motion–Detection and Image–Recording Software Using a Webcam for Application in Business Organization Security

Witthaya Boonsuk^{1*}, Jim Yuennan¹ and Watthana Sriwarom¹

¹ Department of Information Technology, Faculty of Management Sciences and Information Technology, Nakhon Phanom University, Nakhon Phanom, 48000

*Corresponding author: witthaya5773@gmail.com

Received: 1 January 2026; Revised: 16 November 2025; Accepted: 20 April 2026

บทคัดย่อ

วัตถุประสงค์ในการวิจัยนี้ 1) พัฒนาซอฟต์แวร์ที่ประยุกต์ใช้ในการเฝ้าระวังองค์กรธุรกิจ 2) สร้างระบบที่สามารถจับภาพนิ่งอัตโนมัติทุก 10 วินาที เพื่อใช้ในการจัดเก็บข้อมูลและตรวจสอบย้อนหลังได้อย่างสะดวก 3) พัฒนากลไกการตรวจจับการเคลื่อนไหวอัตโนมัติและบันทึกวิดีโอเฉพาะเมื่อมีการเคลื่อนไหวโดยบันทึกต่อเนื่องนาน 30 วินาที ก่อนหยุดการทำงานจนกว่าจะตรวจพบการเคลื่อนไหวครั้งใหม่ 4) เพื่อทดสอบและเปรียบเทียบประสิทธิภาพการประหยัดพื้นที่จัดเก็บข้อมูลของซอฟต์แวร์ที่พัฒนาขึ้น

งานวิจัยนี้ใช้เทคนิคการประมวลผลภาพดิจิทัล (Image Processing) ร่วมกับการตรวจจับการเคลื่อนไหวด้วยวิธี Frame Differencing และ Background Subtraction โดยพัฒนาระบบด้วยภาษา C# เชื่อมต่อกับ Webcam เพื่อบันทึกภาพนิ่งตามช่วงเวลาและวิดีโอเมื่อมีการเคลื่อนไหว พร้อมทั้งจัดเก็บข้อมูลในรูปแบบไฟล์ดิจิทัลเพื่อการเรียกดูย้อนหลัง

ผลการประเมินประสิทธิภาพของซอฟต์แวร์ที่ได้พัฒนาจากการทดสอบบันทึกภาพจาก Webcam ของซอฟต์แวร์ที่พัฒนาซึ่ง ฟังก์ชันที่ 1) ถ่ายภาพนิ่ง ประหยัดพื้นที่ได้ 88.15% ฟังก์ชันที่ 2) ถ่ายวิดีโอขณะมีวัตถุเคลื่อนไหว ประหยัดพื้นที่ได้ 98.75 % ถือว่ามีประสิทธิภาพอยู่ในระดับดีมาก แสดงว่าประสิทธิภาพในการประหยัดพื้นที่ในการบันทึกข้อมูลอยู่ในระดับที่ค่อนข้างดีมาก และมีความเหมาะสมต่อการนำไปประยุกต์ใช้ในงานการตรวจตราและเฝ้าระวังองค์กรธุรกิจ

คำสำคัญ: การตรวจจับภาพเคลื่อนไหว, การประมวลผลภาพ, การเพิ่มประสิทธิภาพการจัดเก็บข้อมูล, ภาพดิจิทัล

¹ สาขาวิชาเทคโนโลยีสารสนเทศ คณะวิทยาการจัดการและเทคโนโลยีสารสนเทศ มหาวิทยาลัยนครพนม จังหวัดนครพนม 48000

Abstract

The objectives of this research are: 1) to develop software applied for the surveillance of business organizations 2) to design a system that can automatically capture still images every 10 seconds, facilitating convenient data storage and retrospective examination 3) to develop an automatic motion detection mechanism that records video only when movement is detected, continuous for 30 seconds before stopping, until new movement is detected. and 4) to test and compare the storage space efficiency of the developed software This approach helps reduce storage usage and enhances practical efficiency.

This research employs digital image processing techniques in combination with motion detection methods, namely frame differencing and background subtraction. The system is developed using C# and connected to a webcam to capture still images at specified intervals and record video only when motion is detected. All data are stored in digital file formats for convenient retrieval and review.

The evaluation results of the developed software, based on webcam recording tests, indicate that: Function 1 (still image capture) achieved 88.15% storage savings. and Function 2 (motion-triggered video recording) achieved 98.75% storage savings. These results demonstrate that the software provides highly effective storage optimization and is suitable for application in monitoring and surveillance of business organizations as well as for further development and commercial.

Keywords: Motion Detection, Image Processing, Storage Optimization, Digital Image

Introduction

ในปัจจุบันมีการนำระบบสารสนเทศเข้ามาประยุกต์ใช้ในการทำงานมากขึ้น และมีบทบาทต่อการทำงานค่อนข้างจะสูงขึ้น ซึ่งมีส่วนจำเป็นอย่างมากในการดำเนินชีวิตในปัจจุบัน ตลอดจนการประยุกต์ในงานด้านธุรกิจ อุตสาหกรรม ด้านวิทยาศาสตร์ ด้านการเกษตร ด้านความปลอดภัย และด้านการผลิต ซึ่งแนวคิดด้านการบริหารจัดการกับการประยุกต์ใช้กล้องรักษาความปลอดภัยในองค์กร ช่วยเสริมการควบคุมภายใน (Internal Control) ทำให้องค์กรสามารถติดตามความเสี่ยงและป้องกันการทุจริตได้อย่างเป็นระบบ และในการจัดการทรัพยากรองค์กร (Resource-Based View: RBV) ซึ่งว่าการใช้เทคโนโลยีกล้องวงจรปิดเป็นทรัพยากรเชิงกลยุทธ์ที่เพิ่มความสามารถในการแข่งขัน (Laudon, et al, 2020)

ปัจจุบัน เทคโนโลยีกล้องดิจิทัลและระบบประมวลผลภาพ (Image Processing) มีบทบาทสำคัญในการเฝ้าระวังและรักษาความปลอดภัยในหลายด้าน ไม่ว่าจะเป็นการติดตั้งกล้องวงจรปิด (CCTV) ในสถานที่ราชการ สถานศึกษา หรือพื้นที่สาธารณะ ตลอดจนการนำมาใช้ในระบบอัจฉริยะต่าง ๆ เช่น บ้านอัจฉริยะ (Smart Home) และการจัดการข้อมูลวิดีโอเพื่อการวิเคราะห์เชิงลึก (Video Analytics) (Szeliski, 2010) การพัฒนาซอฟต์แวร์ที่สามารถบันทึกวิดีโอและจับภาพจาก Webcam จึงเป็นอีกแนวทางที่ได้รับความนิยม เนื่องจากมีต้นทุนต่ำ ติดตั้งง่าย และสามารถพัฒนาต่อยอดให้สอดคล้องกับการใช้งานที่หลากหลาย (Gonzalez & Woods, 2018)

อย่างไรก็ตาม การบันทึกวิดีโอแบบต่อเนื่องอาจทำให้สิ้นเปลืองพื้นที่จัดเก็บข้อมูลจำนวนมาก รวมถึงเพิ่มภาระในการประมวลผลและการตรวจสอบย้อนหลัง (Rad, 2017) ดังนั้น การเพิ่มกลไกการตรวจจับวัตถุเคลื่อนไหว (Motion Detection) เพื่อบันทึกวิดีโอเฉพาะช่วงเวลาที่มีการเปลี่ยนแปลงในภาพ จึงเป็นแนวทางที่ช่วยลดการใช้ทรัพยากร และยังตอบสนองต่อความต้องการใช้งานจริงได้อย่างมีประสิทธิภาพ (Piccardi, 2004)

การวิจัยและพัฒนาซอฟต์แวร์เพื่อตรวจจับวัตถุเคลื่อนไหวที่แบบอัตโนมัติ และบันทึกด้วยกล้อง webcam เป็นเรื่องที่มีความสำคัญและตอบโจทยความต้องการในยุคปัจจุบัน ในด้านรักษาความปลอดภัย ไม่ว่าจะเป็นในแง่ของการลดต้นทุน เพิ่มความเชื่อมั่นหรือความปลอดภัยและเผื่อระวังในองค์กรธุรกิจ นอกจากนี้ยังช่วยสร้างนวัตกรรมใหม่ ๆ ในการวิเคราะห์ข้อมูลด้านความปลอดภัยและการปรับปรุงกระบวนการในภาคอุตสาหกรรมความปลอดภัย ซึ่งเป็นก้าวสำคัญต่อการพัฒนานวัตกรรมในอนาคต (Ratcliffe, 2011) ดังนั้น งานวิจัยนี้จึงมีความสำคัญในการนำเสนอซอฟต์แวร์ที่ผสานทั้งความสามารถในการบันทึกวิดีโอ การจับภาพนิ่ง และการตรวจจับวัตถุเคลื่อนไหวเข้าด้วยกัน อันจะเป็นประโยชน์ต่อการพัฒนาระบบรักษาความปลอดภัย ระบบตรวจสอบอัตโนมัติในองค์กรธุรกิจ รวมถึงสามารถต่อยอดสู่การพัฒนาในเชิงพาณิชย์หรือด้านอื่น ๆ ในอนาคตได้

Objective

1. พัฒนาซอฟต์แวร์ที่สามารถบันทึกภาพจาก Webcam เป็นวิดีโอได้อย่างต่อเนื่องและมีประสิทธิภาพ เพื่อให้ได้ข้อมูลที่ครบถ้วนสำหรับการเฝ้าระวัง
2. สร้างระบบที่สามารถจับภาพนิ่งอัตโนมัติทุก 10 วินาที เพื่อใช้ในการจัดเก็บข้อมูลและตรวจสอบย้อนหลังได้อย่างสะดวก
3. พัฒนากลไกการตรวจจับการเคลื่อนไหวอัตโนมัติ และบันทึกวิดีโอเฉพาะเมื่อมีการเคลื่อนไหว โดยบันทึกต่อเนื่องนาน 30 วินาที ก่อนหยุดการทำงานจนกว่าจะตรวจพบการเคลื่อนไหวครั้งใหม่
4. เพื่อทดสอบและเปรียบเทียบประสิทธิภาพการประหยัดพื้นที่จัดเก็บข้อมูลของซอฟต์แวร์ที่พัฒนาขึ้น

Literature Review

การพัฒนาโปรแกรมบันทึกภาพจากกล้อง CCTV หรือ Webcam ที่สามารถตรวจจับการเคลื่อนไหวและบันทึกได้ มักอาศัยการประมวลผลภาพเพื่อแยกความแตกต่างระหว่างฉากพื้นหลังกับวัตถุที่เคลื่อนไหว (Background Subtraction) ซึ่งช่วยลดข้อมูลที่ไม่เกี่ยวข้องและเพิ่มประสิทธิภาพการตรวจจับ (Piccardi, 2004) เทคนิค Gaussian Mixture Models (GMM) เป็นหนึ่งในวิธีที่ได้รับความนิยมในการสร้างโมเดลพื้นหลังที่ยืดหยุ่นต่อการเปลี่ยนแปลงของแสง (Zivkovic, 2004) ระบบตรวจจับการเคลื่อนไหวมักใช้การหาระยะเปลี่ยนแปลงของพิกเซลระหว่างเฟรม (Frame Differencing) ร่วมกับการกรองเสียงรบกวน (Noise Filtering) เพื่อให้สามารถตัดสินใจได้ว่าเป็นการเคลื่อนไหวที่สำคัญหรือไม่ (Toyama et al., 1999)

นอกจากนี้การใช้การตรวจจับขอบ (Edge Detection) และคุณลักษณะของวัตถุช่วยเพิ่มความแม่นยำในการแยกวัตถุจากพื้นหลัง (Canny, 1986) ระบบบันทึกภาพอัตโนมัติยังต้องพิจารณาเรื่องการจัดเก็บข้อมูลอย่างมีประสิทธิภาพ เช่น การบีบอัดวิดีโอ (Video Compression) เพื่อประหยัดพื้นที่จัดเก็บและเพิ่มความเร็วในการเข้าถึงข้อมูลย้อนหลัง (Wiegand et al., 2003) การผสานเทคโนโลยี Deep Learning ล่าสุด เช่น Convolutional Neural Networks (CNNs) ช่วยให้ระบบตรวจจับเคลื่อนไหวสามารถจำแนกวัตถุและบริบทได้ดีขึ้น (Redmon et al., 2016)

งานวิจัยนี้ใช้เทคนิคการประมวลผลภาพดิจิทัล (Image Processing) ร่วมกับการตรวจจับการเคลื่อนไหวด้วยวิธี Frame Differencing และ Background Subtraction โดยพัฒนาระบบด้วยภาษา C# เชื่อมต่อกับ Webcam เพื่อบันทึกภาพนิ่งตามช่วงเวลาและวิดีโอเมื่อมีการเคลื่อนไหว พร้อมทั้งจัดเก็บข้อมูลในรูปแบบไฟล์ดิจิทัลเพื่อการเรียกดูย้อนหลัง

Methodology

ความสำคัญของ CCTV

เนื่องจากปัจจุบันโลกมีแนวโน้มที่จะนำระบบอัตโนมัติเข้ามาใช้งานและประยุกต์ใช้งานมากขึ้น แต่ยังคงขาดแคลนระบบประมวลผลอัตโนมัติ การนำระบบกล้องโทรทัศน์วงจรปิด (Closed-Circuit Television: CCTV) มาใช้ในองค์กรธุรกิจถือเป็นหนึ่งในเทคโนโลยีที่มีบทบาทสำคัญต่อการบริหารจัดการด้านความปลอดภัยในยุคดิจิทัล โดยหลักการของ CCTV คือการบันทึกภาพเหตุการณ์แบบเรียลไทม์ผ่านเครือข่ายปิดสำหรับการตรวจสอบ การเฝ้าระวัง และการเก็บหลักฐาน ซึ่งสอดคล้องกับแนวคิดการบริหารความเสี่ยง (Risk Management) ที่เน้นการลดโอกาสในการเกิดเหตุ และเพิ่มความสามารถในการตอบสนองเหตุการณ์ผิดปกติอย่างทันท่วงที (Gill, et al., 2005), (Norris, et al., 2004) หลักการบริหารความเสี่ยง (Risk Management) มีการสนับสนุนการใช้กล้องเพื่อระบุ วิเคราะห์ และลดผลกระทบของเหตุการณ์ด้านความปลอดภัย ในระบบสารสนเทศเพื่อการจัดการ (Management Information Systems: MIS) กล้องสามารถสร้างข้อมูลที่ช่วยผู้บริหารตัดสินใจเชิงปฏิบัติการและเชิงกลยุทธ์ได้ และแนวคิดความปลอดภัยเชิงป้องกัน (Preventive Security) ระบุว่าควรมีกองเป็นมาตรการเชิงรุกที่ลดโอกาสเกิดเหตุและเพิ่มความมั่นใจแก่บุคลากร (Laudon, et al., 2020), (Robbins, et al., 2018), (Barney 1991)

ขั้นตอนกระบวนการพัฒนาและการวิจัย

การวิจัยครั้งนี้เป็นการวิจัยและพัฒนา (Research and Development: R&D) ที่มุ่งเน้นการสร้างซอฟต์แวร์สำหรับตรวจจับวัตถุเคลื่อนไหวและบันทึกภาพจากกล้อง Webcam โดยมีขั้นตอนการดำเนินงานดังนี้

1. การศึกษาและวิเคราะห์ความต้องการ (Requirement Analysis)

ปัจจุบันได้มีการนำกล้อง CCTV มาบันทึกภาพเหตุการณ์แบบเรียลไทม์ผ่านเครือข่ายปิดสำหรับการตรวจสอบ แต่ยังคงขาดความหลากหลายในการทำงานจึงจำเป็นต้องมีการประยุกต์เอาเทคโนโลยีการประมวลผลภาพและสำหรับตรวจจับการเคลื่อนไหวและควบคุมการบันทึกวิดีโออัตโนมัติมาใช้งาน ซึ่งต้องมีการศึกษาดังนี้

1.1 ศึกษาข้อมูลและงานวิจัยที่เกี่ยวข้องกับเทคโนโลยีการประมวลผลภาพ (Image Processing) และการตรวจจับการเคลื่อนไหว (Motion Detection)

1.2 วิเคราะห์ความต้องการของผู้ใช้งานจริง เช่น ระบบรักษาความปลอดภัย การตรวจสอบการเข้า-ออกพื้นที่ หรือการเก็บข้อมูลอัตโนมัติ กำหนดคุณสมบัติหลักของระบบให้สอดคล้องกับวัตถุประสงค์ ได้แก่

- 1) การบันทึกวิดีโอจากกล้อง Webcam ได้ต่อเนื่อง
- 2) การจับภาพนิ่งอัตโนมัติทุก 10 วินาที
- 3) การบันทึกวิดีโอเฉพาะเมื่อมีการตรวจจับวัตถุเคลื่อนไหว

2. การออกแบบระบบ (System Design)

การออกแบบได้ทำการออกแบบระบบมีส่วนการทำงานโดยออกแบบสถาปัตยกรรมของระบบ โดยแบ่งออกเป็น 3 ส่วน การออกแบบฐานข้อมูลหรือโครงสร้างการจัดเก็บไฟล์ และวางแผนการใช้เครื่องมือและภาษาโปรแกรม

2.1 ออกแบบสถาปัตยกรรมของระบบ โดยแบ่งออกเป็น 3 โมดูลหลัก ได้แก่

- 1) Video Recording Module – สำหรับบันทึกวิดีโอจาก Webcam
- 2) Image Capture Module – สำหรับจับภาพนิ่งตามช่วงเวลา
- 3) Motion Detection Module – สำหรับตรวจจับการเคลื่อนไหวและควบคุมการบันทึกวิดีโออัตโนมัติ

2.2 ออกแบบฐานข้อมูลหรือโครงสร้างการจัดเก็บไฟล์ ทั้งในรูปแบบวิดีโอและภาพนิ่ง

2.3 วางแผนการใช้เครื่องมือและภาษาโปรแกรม เช่น C# และ OpenCV บน .NET Framework เพื่อรองรับการพัฒนา

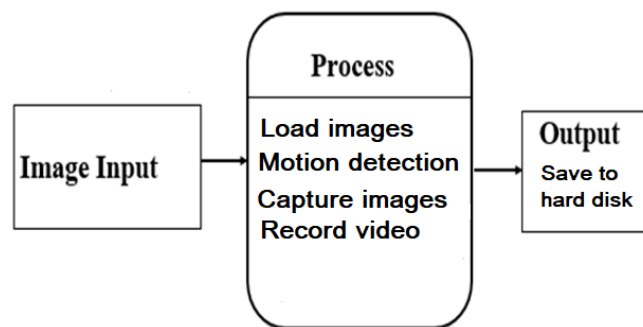


Figure 1 Overview of the context diagram of the system

จาก Figure 1 แสดงการทำงานของระบบ โดยระบบจะทำการรับข้อมูลภาพจากกล้อง webcam แล้วทำการส่งเข้าสู่ระบบเพื่อทำการประมวลผลจับการเคลื่อนไหวของภาพและทำการบันทึกจัดเก็บตามขั้นตอนที่มีการประมวลของระบบ และส่งออกไฟล์ข้อมูลภาพลงสู่สื่อจัดเก็บ (Hard disk)

โครงสร้างของระบบแสดงใน Figure 2

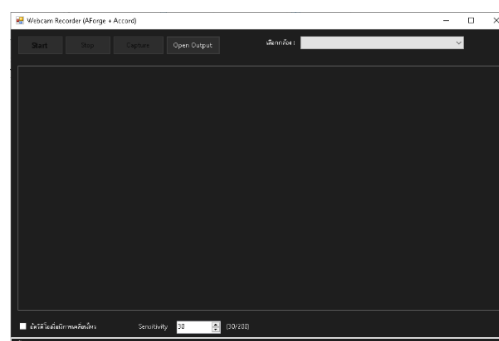


Figure 2 System structure

จากรูปที่ 2 โครงสร้างระบบจะประกอบไปด้วย 3 ส่วนคือ

1. Hardware ประกอบไปด้วย คอมพิวเตอร์และกล้องบันทึกภาพนิ่งความละเอียดสูง
2. Software คือระบบที่เราได้ออกแบบและพัฒนา

ขั้นตอนการทำงานของระบบแสดงใน Figure 3

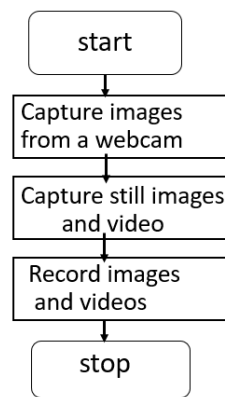


Figure 3 System processing

จาก Figure 3 ขั้นตอนของระบบมีรายละเอียดการทำงานดังนี้

1. (Image Input) รับข้อมูลภาพจากกล้อง webcam แล้วทำการส่งเข้าสู่ระบบเพื่อทำการประมวลผล
2. (Capture Image and Video) จับภาพนิ่งและภาพวิดีโอ
3. (Record Image and Video) บันทึกภาพนิ่งและภาพวิดีโอ

3. การพัฒนาระบบ (System Development)

การพัฒนาระบบได้ทำการเลือกเครื่องมือในการพัฒนาด้วยภาษา C# ออกแบบสถาปัตยกรรมของระบบ โดยแบ่งออกเป็น 2 ส่วน พัฒนาโปรแกรมด้วยภาษา C# และติดตั้งไลบรารีที่เกี่ยวข้อง ไฟล์ และพัฒนาและทดสอบแต่ละโมดูล ตลอดจนใช้เครื่องมือและภาษาโปรแกรม

3.1 พัฒนาโปรแกรมด้วยภาษา C# และติดตั้งไลบรารีที่เกี่ยวข้อง เช่น AForge.NET หรือ Emgu CV และ Accord.Video.FFMPEG สำหรับการประมวลผลภาพ

ขั้นตอนวิธีการรหัสเทียมหรือ (Coding) ของระบบ

```

using System;
using System.Drawing;
using System.IO;
using System.Windows.Forms;
using AForge.Video;
using AForge.Video.DirectShow;
using Accord.Video.FFMPEG;
namespace WebcamRecorder
{
    public partial class Form1 : Form
    {
        private FilterInfoCollection videoDevices;
        private VideoCaptureDevice videoSource;
        private VideoFileWriter videoWriter;
        private bool isRecording = false;
    }
  
```

```

private DateTime lastMotionTime;
private Timer snapshotTimer;
private Timer motionTimer;
private Bitmap lastFrame;
public Form1()
{
    InitializeComponent(); // Timer สำหรับถ่ายภาพนิ่งทุก 10 วินาที
    snapshotTimer = new Timer();
    snapshotTimer.Interval = 10000;
    snapshotTimer.Tick += SnapshotTimer_Tick; // Timer สำหรับบันทึกวิดีโอเมื่อมีการ
เคลื่อนไหว
    motionTimer = new Timer();
    motionTimer.Interval = 10000; // บันทึก 10 วินาที
    motionTimer.Tick += MotionTimer_Tick;
}

```

3.2 พัฒนาและทดสอบแต่ละโมดูล ได้แก่

1. การบันทึกวิดีโอ ให้ทำงานได้ต่อเนื่องโดยไม่กระตุก
2. การจับภาพนิ่งอัตโนมัติทุก 10 วินาที พร้อมบันทึกลงไฟล์เดอร์
3. การตรวจจับการเคลื่อนไหว โดยใช้เทคนิคการเปรียบเทียบภาพระหว่างเฟรม (Frame

Differencing) หรือ Background Subtraction เพื่อหาความเปลี่ยนแปลงของภาพ และบันทึกวิดีโอเมื่อมีการเคลื่อนไหว

การพัฒนาหน้าจอการทำงาน (User Interface) แสดงใน Figure 4

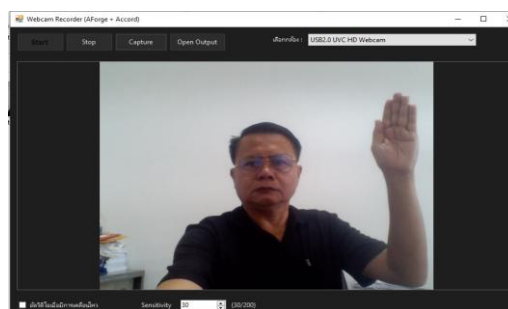


Figure 4 Prototype Screen

4. การทดสอบระบบ (UnterkaImsteiner, et al. 2023)

การวัดประสิทธิภาพของระบบสามารถทำได้โดยการใช้วิธี วัดค่าความแม่นยำ (Precision) ซึ่งเป็นการวัดความแม่นยำในการจัดเก็บภาพนิ่งและภาพวิดีโอ เทียบกับการใช้พื้นที่ในหน่วยความจำปกติที่บันทึกในระบบที่ประยุกต์

โดยพิจารณาจากจำนวนพื้นที่ในการจัดเก็บที่เปรียบเทียบจากสื่อบันทึกของมาตรฐานกลางทั่วไป ที่ใช้เป็นข้อมูลเปรียบเทียบในการใช้ความจำทั้งหมดว่ามีภาพหรือวิดีโอที่บันทึกทั่วไป นำมาทดสอบเทียบกับภาพหรือวิดีโอของระบบที่พัฒนานำมาทดสอบและทำการบันทึกกับกล้องวิดีโอที่บันทึกทั่วไป ซึ่งจะมีค่าหน่วยความจำที่จัดเก็บใช้พื้นที่แตกต่างกันมากน้อยกับระบบที่ได้พัฒนาและนำมาคำนวณ แสดงดังสมการ

$$\text{Precision} = \frac{|x_i - x_m|}{x_m}$$

$$x_m = \frac{1}{n} \sum_{i=1}^n x_i$$

โดยที่ x_m คือ ค่าเฉลี่ยของการวัด

x_i คือ ค่าการวัดแต่ละครั้ง

Results

จากผลการออกแบบและพัฒนาระบบที่ได้ ทำให้มีระบบต้นแบบนำมาทดสอบการวัดประสิทธิภาพของระบบสามารถทำได้โดยการใช้วิธี วัดค่าความแม่นยำ (Precision) ซึ่งเป็นการวัดความแม่นยำในการจัดเก็บภาพนิ่งและภาพวิดีโอ เทียบกับการใช้พื้นที่ในหน่วยความจำปกติที่บันทึกในระบบที่ประยุกต์

การกำหนดรายละเอียดในการทดสอบจากกล้อง webcam มาตรฐานกับห้องสภาพแวดล้อมแสงปกติ กลางวัน ยกเว้นกล้องพิเศษ เช่นกล้องที่สามารถทำงานในกลางคืนได้ ก็เพิ่มประสิทธิภาพการทำงานในกลางคืนได้ดี

1. ผลการทดสอบและพัฒนาระบบหน้าจอกการทำงาน (User Interface) แสดงใน Figure 5

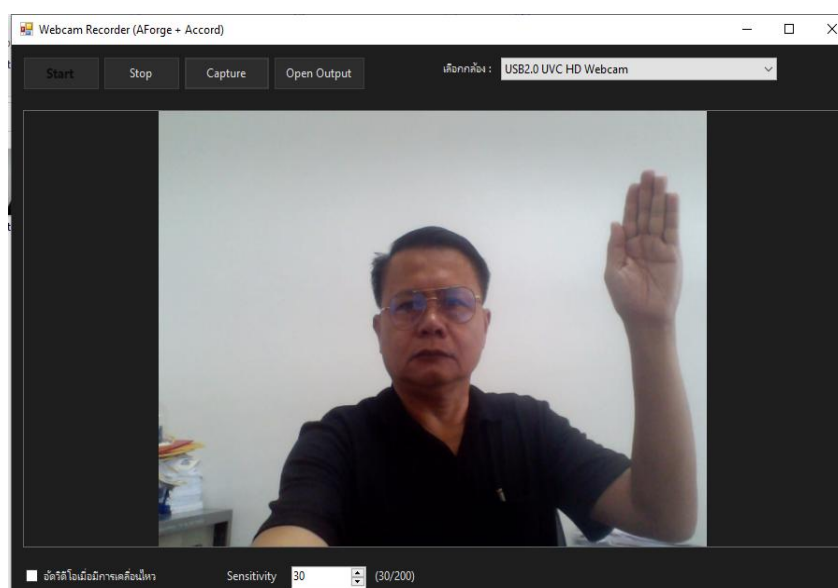


Figure 5 Software Prototype

2. ผลการประเมินประสิทธิภาพของระบบ

การประเมินหาประสิทธิภาพ ของระบบตรวจสอบการใช้พื้นที่ในการจัดเก็บในหน่วยความจำทั้งภาพนิ่งและวิดีโอในเวลา 1 วัน หรือ 24 ชั่วโมง ในการเปรียบเทียบการใช้ความจำและวิเคราะห์ผลต่างของพื้นที่จัดเก็บ ใช้ต่างกันอย่างไร จากภาพนิ่งที่บันทึกและวิดีโอที่บันทึก กรณีทดสอบจากกลุ่มตัวอย่างภาพนิ่งและวิดีโอ แสดงค่าเฉลี่ยในเชิงปริมาณ และค่าเฉลี่ยในเชิงคุณภาพที่นำมาทดสอบในช่วงเวลา 24 ชั่วโมง แบ่งเป็น 2 (Function Test)

กลุ่มที่ 1 ภาพนิ่งที่บันทึกใน 1 วัน

กลุ่มที่ 2 ภาพวิดีโอที่บันทึกใน 1 วัน

หลังจากที่ได้นำซอฟต์แวร์ ไปทดสอบตามวิธีการแบบ(Black Box) เรียบร้อยแล้วต่อไปจะเป็นการนำซอฟต์แวร์นี้ไปประเมิน เพื่อหาประสิทธิภาพของระบบซอฟต์แวร์ และเป็นการทดสอบ เพื่อยอมรับระบบโดยผู้ใช้งาน (Acceptance Test by User) (Thaweepol, 2012)

ซึ่งกระบวนการประเมินระบบนี้ เป็นการประเมินเพื่อหาประสิทธิภาพของผลงานทางระบบสารสนเทศ และซอฟต์แวร์ซึ่งจะแบ่งการประเมินระบบออกเป็น 4 ส่วนด้วยกันคือ 1) Function Requirement Test 2) Function Test 3) Usability Test และ 4) Security Test

แต่ในกรณีนี้ ต้องการทดสอบระบบหาประสิทธิภาพของระบบหรือซอฟต์แวร์อย่างเดียว จึงเลือกใช้การวัดประสิทธิภาพของโปรแกรม โดยใช้เกณฑ์ทดสอบทางประสิทธิภาพของการทำงานของซอฟต์แวร์คือ แบบทดสอบทาง (Function Test) สำหรับการเตรียมข้อมูลที่ใช้โดยเป็นภาพดิจิทัลสไลด์ทั่วไป ซึ่งในการทดลองนี้ได้ทดสอบกับข้อมูลที่ได้จากการบันทึกภาพวิดีโอจากกล้องมาตรฐานทั่วไป ซึ่งภาพนิ่งบันทึกเป็น JPEG ขนาดเฉลี่ย 300 KB ต่อภาพและถ้าถ่ายภาพวิดีโอ กล้องเว็บแคมธรรมดา 720 p การถ่ายวิดีโอ 1 วินาที จะใช้พื้นที่ประมาณ 0.25 MB/s

ผลการคำนวณพื้นที่จัดเก็บข้อมูลจากกล้องเว็บแคม

1. การถ่ายภาพนิ่งทุก 10 วินาทีในเวลา 24 ชั่วโมงหรือ 1 วัน ด้วยกล้องเว็บแคมธรรมดาทั่วไป จะใช้พื้นที่ความจุกี่ไบต์ (Logitech, n.d. 2023)

1.1 จำนวนภาพที่ได้ใน 1 วัน 1 วัน = 24 ชั่วโมง 1 ชั่วโมง = 3600 วินาที

รวม = $24 \times 3600 = 86,400$ วินาที

ถ่ายทุก 10 วินาที $\rightarrow 86,400 \div 10 = 8,640$ ภาพ $86,400 \div 10 = 8,640$ ภาพ

1.2 ขนาดไฟล์ภาพจากกล้องเว็บแคมทั่วไป

ความละเอียดเว็บแคมทั่วไป: 640×480 (VGA) ถึง 1280×720 (HD) หากบันทึกเป็น JPEG ขนาดเฉลี่ย ~ 200 KB – 500 KB ต่อภาพ เพื่อความสะดวก เอาค่าเฉลี่ย 300 KB ต่อภาพ

1.3 พื้นที่รวมที่บันทึกได้ใน 1 วัน

$8,640 \times 300 \text{ KB} = 2,592,000 \text{ KB}$ $8,640 \times 300 \text{ KB} = 2,592,000 \text{ KB}$

แปลงเป็น MB (1 MB = 1024 KB):

$2,592,000 \div 1024 \approx 2,531 \text{ MB}$ $2,592,000 \div 1024 \approx 2,531 \text{ MB}$

ใช้พื้นที่รวมในการบันทึกได้ใน 1 วัน $\approx 2.5 \text{ GB}$ ต่อวัน

2. การบันทึกวิดีโอทั่วไป (Wikipedia., 2023)

2.1 กล้องเว็บแคมธรรมดา ส่วนใหญ่ ความละเอียด 720p (1280×720) 30 fps

รูปแบบการบันทึกวิดีโอทั่วไป MP4 (H.264) บิตเรต (Bitrate) ของเว็บแคม 1) ต่ำ (VGA/480p): ~1 Mbps 2) ปานกลาง (720p): ~2–3 Mbps 3) สูง (1080p): ~4–6 Mbps (1 Mbps = 1 Megabit ต่อวินาที = 0.125 MB/s เพราะ 1 Byte = 8 bits)

2.2. พื้นที่เก็บข้อมูล (ตัวอย่าง 720p ~ 2 Mbps)

1) 2 Mbps = $2 \div 8 = 0.25$ MB/s

2) ความยาววิดีโอ 30 วินาที $\rightarrow 0.25 \times 30 = 7.5$ MB $0.25 \times 30 = 7.5$ MB

ดัง Table 1 เปรียบเทียบ Bitrate โดยประมาณตามความละเอียดของกล้อง การใช้พื้นที่จัดเก็บในหน่วยความจำของการบันทึกไฟล์รูปแบบ วิดีโอ ใน 1 วินาที และ 30 วินาที จากกล้องที่มีความละเอียด 720p (HD) ซึ่งในการทดลองใช้ความละเอียดที่ 720p (HD) Bitrate โดยประมาณ (Digital Samba, 2023), (Wowza , 2023).

Table 1 ตารางเปรียบเทียบ Bitrate โดยประมาณ ความละเอียดของกล้อง

Approximate	Bitrate	Storage (MB/s)	Storage (MB/30s)
480p (VGA)	1 Mbps	0.125 MB/s	~3.75 MB
720p (HD)	2–3 Mbps	0.25–0.375 MB/s	~7.5–11 MB
1080p (Full HD)	4–6 Mbps	0.5–0.75 MB/s	~15–22.5 MB

ดัง Table 2 จากการทดลองเปรียบเทียบซอฟต์แวร์ที่พัฒนากับกล้อง CCTV ทั่วไป ซึ่งซอฟต์แวร์ที่พัฒนามีฟังก์ชันการทำงาน 3 แบบ คือ 1)บันทึกภาพนิ่ง 2)บันทึกวิดีโอขณะมีวัตถุเคลื่อนไหว 3)บันทึกวิดีโอตลอด 24 ชั่วโมง จากการทดสอบของซอฟต์แวร์ ฟังก์ชันที่ 3 มีการจัดบันทึกของหน่วยความจำ เท่ากันกับ กล้อง CCTV ทั่วไป จึงไม่ทำการเปรียบเทียบ จึงใช้แค่ 2 ฟังก์ชันของซอฟต์แวร์ที่พัฒนาเท่านั้นในการเปรียบเทียบ

Table 2 เปรียบเทียบซอฟต์แวร์ที่พัฒนากับกล้อง CCTV ทั่วไป

Comparative experiments of the developed software with conventional CCTV cameras	Functionality	Storage /day (GB)	Storage /day (GB)
1. Software developed using a web cam	1.1 Take a still photo every 10 seconds, once a day.	8,640 images/day $\times$ 300 KB/ images = 2,592,000 KB	2.5 GB/ day
	1.2 Record video while there is movement.	MB per 30 seconds 11 MB	It depends on the number of times an object passes by.
	1.3 Record video 24 hours a day.	One day has 86,400 seconds (approximate video	

Comparative experiments of the developed software with conventional CCTV cameras	Functionality	Storage /day (GB)	Storage /day (GB)
		bitrate 0.25 MB/s).=86,400 × 0.25 = 21,600 MB = 21,600/1024 = 21.09 GB	21.09 GB/day
2.General CCTV cameras	1. Record video 24 hours a day.	One day has 86,400 seconds (approximate video bitrate 0.25 MB/s).=86,400 × 0.25 = 21,600 MB = 21,600/1024 = 21.09 GB	21.09 GB/day

Table 3 พื้นที่บันทึกของซอฟต์แวร์ที่พัฒนาทำการเปรียบเทียบกับกล้อง CCTV ทั่วไปที่ต้องใช้พื้นที่ 21.09 GB/วัน

Comparative experiments of the developed software with conventional CCTV cameras	Functionality	Storage space/day (GB)	Calculated as a % of general CCTV cameras/day 21.09 (GB)
1.Software Developed Using a Web Camera	1.1 Still photography	2.5 GB/day	11.85 %
	1.2 Recording video while objects are moving is calculated at 1 hour per time (11 MB × 24)=264 MB.	0.264 GB/วัน	1.25 %
2. General CCTV Cameras	video recording	21.09 GB/day	100 %

Conclusion and Discussion

จาก Table 2 และ Table 3 การทดสอบประสิทธิภาพ โดยการวัดค่าการจัดเก็บข้อมูลจากซอฟต์แวร์ต้นแบบที่พัฒนา และนำมาทดสอบกับกล้อง CCTV ทั่วไป กับฟังก์ชันการทำงานที่พัฒนาขึ้นทั้ง 3 ฟังก์ชัน แต่ที่นำมาทดสอบทั้งหมด 2 ฟังก์ชันคือ 1) บันทึกภาพนิ่ง 2) บันทึกวิดีโอขณะมีวัตถุเคลื่อนไหว โดยฟังก์ชันที่ 3 ไม่ทดลองเพราะพื้นที่ในการจัดเก็บในการบันทึกมีค่าเท่ากับกล้อง CCTV ทั่วไป โดยใช้การทดสอบในเวลา 1 วัน หรือ 24 ชั่วโมง เพื่อหาประสิทธิภาพการจัดเก็บที่ถูกต้อง จากการบันทึกด้วยกล้อง CCTV ทั่วไป และกล้อง Web cam จากซอฟต์แวร์ที่พัฒนา

โดยผลการวัดค่าการจัดเก็บในการทดสอบมีดังนี้ ฟังก์ชันคือ 1) บันทึกภาพนิ่ง ใช้พื้นที่จัดเก็บ 2.5 GB/วัน คิดเป็น 11.85 % ของการบันทึกวิดีโอ จากกล้อง CCTV ทั้งหมด 2) บันทึกวิดีโอขณะมีวัตถุเคลื่อนไหว คิดที่ 1 ชั่วโมงต่อครั้ง $(11 \text{ MB} \times 24) = 264 \text{ MB}$ ใช้พื้นที่จัดเก็บ 0.264 GB/วัน คิดเป็น 1.25 % ของการบันทึกวิดีโอ จากกล้อง CCTV ทั้งหมด 3) ถ่ายวิดีโอด้วยกล้อง CCTV ทั้งหมด และซอฟต์แวร์ที่พัฒนาโดยใช้กล้อง web cam ในเวลา 24 ชั่วโมงเท่ากันใช้พื้นที่จัดเก็บ 21.09 GB/วัน คิดเป็น 100 % ของการบันทึกวิดีโอ ในเวลา 24 ชั่วโมง

การพัฒนาซอฟต์แวร์ต้นแบบที่ใช้ในการจัดเก็บภาพถ่ายและวิดีโอด้วยกล้อง CCTV ทั้งหมดทำการพัฒนาด้วยภาษา C# โดยใช้เป็นเครื่องมือในการพัฒนาซอฟต์แวร์ เพื่อตรวจสอบประสิทธิภาพความแม่นยำและการใช้พื้นที่บันทึกข้อมูล เมื่อทำการเปรียบเทียบกับกล้อง CCTV ทั้งหมด กับข้อมูลที่ได้จากการประมวลผลของระบบที่พัฒนาขึ้นใหม่ จากการทดสอบในเวลา 24 ชั่วโมง จากซอฟต์แวร์ต้นแบบที่พัฒนา และนำมาทดสอบกับกล้อง CCTV ทั้งหมด ซึ่งนำมาทดสอบทั้งหมด 2 ฟังก์ชันคือ 1) บันทึกภาพนิ่ง 2) บันทึกวิดีโอขณะมีวัตถุเคลื่อนไหว ซึ่งใช้พื้นที่ 11.85% และ 1.25% ของกล้อง CCTV ทั้งหมด ตามลำดับ ซึ่งฟังก์ชันที่ใช้คือ 1) บันทึกภาพนิ่ง ประหยัดพื้นที่ได้ 88.15 % ฟังก์ชันที่ใช้คือ 2) บันทึกวิดีโอขณะมีวัตถุเคลื่อนไหว ประหยัดพื้นที่ได้ 98.75 % ถือว่ามีประสิทธิภาพอยู่ในระดับดีมาก แสดงว่าประสิทธิภาพในการประหยัดพื้นที่ในการบันทึกข้อมูล อยู่ในระดับที่ค่อนข้างดีมาก และมีความเหมาะสมต่อการนำไปประยุกต์ใช้ในงานการตรวจตราและเฝ้าระวัง อีกทั้งมีความสะดวกในการใช้งานและลดต้นทุนเพราะสามารถติดตั้งในคอมพิวเตอร์ส่วนบุคคลและกล้อง Web cam ทั้งหมดได้ ซึ่งจะสามารถนำระบบที่พัฒนาไปประยุกต์ใช้งานเฝ้าระวังในองค์กรธุรกิจ ให้เกิดประสิทธิภาพต่อไป

Acknowledgements

งานวิจัยฉบับนี้สำเร็จลงได้ด้วยดี เนื่องจากได้รับความกรุณาอย่างสูงจากคณบดี ทีมผู้บริหารคณะทุกท่าน ที่กรุณาให้การสนับสนุนทุนทำวิจัย คำแนะนำให้คำปรึกษา ตลอดจนสถานที่ทำวิจัย ห้องปฏิบัติการของคณะ ด้วยความเอาใจใส่อย่างยิ่ง ผู้วิจัยตระหนักถึง ความตั้งใจจริงและความทุ่มเทของผู้บริหารและขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

References

- Barney, J. (1991). Firm resources and sustained competitive advantage. *Journal of Management*, 17(1), 99–120.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6), 679–698.
- Digital Samba. (2023). *Unravelling video bitrate: How to boost video quality and streaming performance*. Digital Samba. Logitech. (n.d.). Logitech C270 HD Webcam. Logitech.
- Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th ed.). Pearson.
- Gill, M., & Spriggs, A. (2005). *Assessing the impact of CCTV*. Home Office Research Study 292.
- Laudon, K. C., & Laudon, J. P. (2020). *Management information systems: Managing the digital firm* (16th ed.). Pearson.
- Norris, C., McCahill, M., & Wood, D. (2004). The growth of CCTV: A global perspective on the international diffusion of video surveillance in publicly accessible space. *Surveillance & Society*, 2(2/3), 110–135.
- Piccardi, M. (2004). Background subtraction techniques: a review. *IEEE International Conference on Systems, Man and Cybernetics*, 4, 3099–3104.

- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 779–788).
- Robbins, S. P., & Coulter, M. (2018). *Management* (14th ed.). Pearson.
- Rad, A. A. (2017). Motion detection for real-time surveillance applications. *International Journal of Computer Applications*, 168(4), 23–28.
- Ratcliffe, J. (2011). *Video surveillance and crime reduction: The effect of CCTV on crime prevention. Policing*, 5(3), 233–239.
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer.
- Toyama, K., Krumm, J., Brumitt, B., & Meyers, B. (1999). *Wallflower: Principles and practice of background maintenance*. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, (1), 255–261.
- Unterkalmsteiner, M., Gorschek, T., Islam, A. K. M. M., Cheng, C.K., Permadi, R. B., & Feldt, R. (2023). *Evaluation and measurement of software process improvement A systematic literature review*.
- Wikipedia. (2023). *Advanced Video Coding*. In Wikipedia.
- Wikipedia. (2023). *Bit rate*. In Wikipedia. Form https://en.wikipedia.org/wiki/Bit_rate
- Wikipedia. (2023). *Data-rate units*. In Wikipedia. Form https://en.wikipedia.org/wiki/Data-rate_units
- Wowza. (2023). *What is video bitrate and what bitrate should you use*. Wowza Media Systems. Form <https://www.wowza.com/blog/what-is-video-bitrate-and-what-bitrate->
- Wikipedia. (2023). *Bit rate*. In Wikipedia. Form https://en.wikipedia.org/wiki/Bit_rate
- Wiegand, T., Sullivan, G. J., Bjøntegaard, G., & Luthra, A. (2003). Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7), 560–576.
- Zivkovic, Z. (2004). Improved adaptive Gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004* (2), pp. 28–31.