

Portable Image Platform (PIP)

Sophie Gerrick, Jeffrey Lee, Joey Degges, Vincent Simoes, Michael Wilson,
Teresa Ko, John Hicks, Eric Graham, Deborah Estrin

UCLA CENS – <http://sites.google.com/site/GumstixCENS/Home>

Introduction: Robust and Efficient Image Sensors

Hardware and Software are readily available and highly robust

Data storage is maximized through selection of meaningful encoding schemes

Hardware

The hardware should be standard, *non-specialized* hardware that can be deployed as fungible imager platforms by researchers *without significant modification* or effort.

Encoding options

Standard video encoding schemes try to maintain a *consistent quality level* across all frames. This results in an equal disbursement of bits throughout the video sequence.

Software

The software should be freely available and *robust* enough for research applications requiring the specified hardware configuration. In addition, camera settings should be easily modified through an *intuitive UI* to match the deployment environment.

Fixed Data Storage

Realistic deployments require data compression methods that maximize representations of *interesting data* while minimizing *noise and redundant information*.

Problem Description: Help researchers efficiently collect image data

Images as Sensors Platform

Use Case Scenario

Researchers seek to identify the frequency of pollination in specific floral regions. Viewing and recording these pollinators can often be a *time-consuming task* for field researchers. Traditional methods require direct, personal observations of single floral regions. PIP is designed to facilitate the process by using a camera to record relevant data that can then be viewed and analyzed. This will *significantly save time* by allowing the researcher to view only the data that relates to the pollinator.

System Description

- *Easily reproducible* data logger
- Robust enough to handle *varied image collection applications*
- Cost effective and economically feasible
- Lower power and highly portable, for *easy field deployment*
- Ability to *remotely change* camera settings
- Light Web-based interface with camera
- Off the shelf components

Proposed Solution: Highly reproducible image data logger with specialized encoding schemes

USB Camera

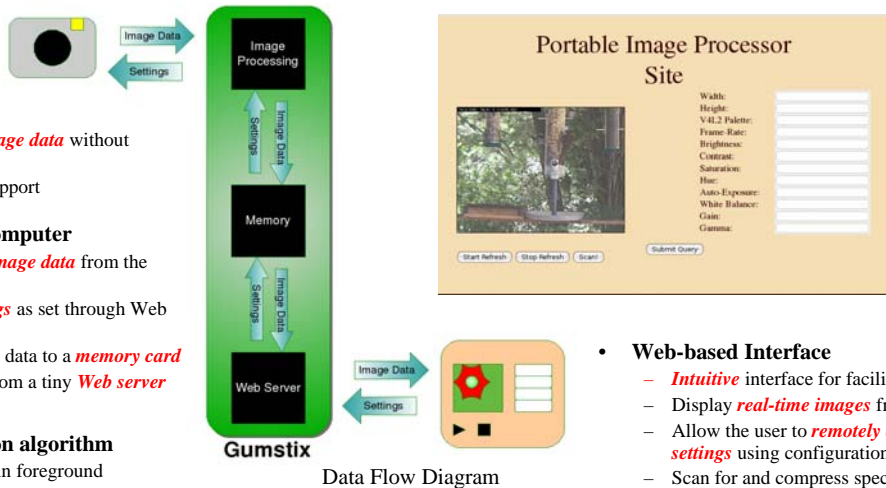
- Reasonable *quality image data* without frame interlace
- Video4Linux driver support

Gumstix mini Linux computer

- *Receive and process image data* from the camera
- Provide *camera settings* as set through Web interface
- Write processed image data to a *memory card*
- Serve Web interface from a tiny *Web server*

Background subtraction algorithm

- Identify *novel objects* in foreground
- *Minimize background* storage requirements
- More *efficient* than the MPEG4 H.264/AVC video encoder, x264.
- *Data quality of novel objects* remains higher than full frame encoding



Web-based Interface

- *Intuitive* interface for facilitated use
- Display *real-time images* from the camera
- Allow the user to *remotely change the camera settings* using configuration files
- Scan for and compress specific directories on the data logger
- Present a *compressed archive of images* for download



Background Subtraction at Birdfeeders



Mpeg4 H.264 Compression



Novel Object-Sensitive Compression