ESP32-CAM As A Programmable Camera Research Platform

Henry Dietz, Dillon Abney, Paul Eberhart,
Nick Santini, Wil Davis, Elisabeth Wilson,
and Michael McKenzie

ISS-232, 14:30, January 19, 2022

University of Kentucky
Electrical & Computer Engineering

University of Kentucky

Programmable Cameras

- Commercial units from Baumer, Fisher Smith, Roseek, etc.
- **DevCAM** (2020-): D. Meyer at UCSD; up to 6 MIPI cameras controlled by Xilinx SoC
- **Raspberry Pi** (2013-): camera modules...
- **OpenMV** (2013-): camera board for machine vision up to 640x480, program in MicroPython

Program Consumer Cameras?

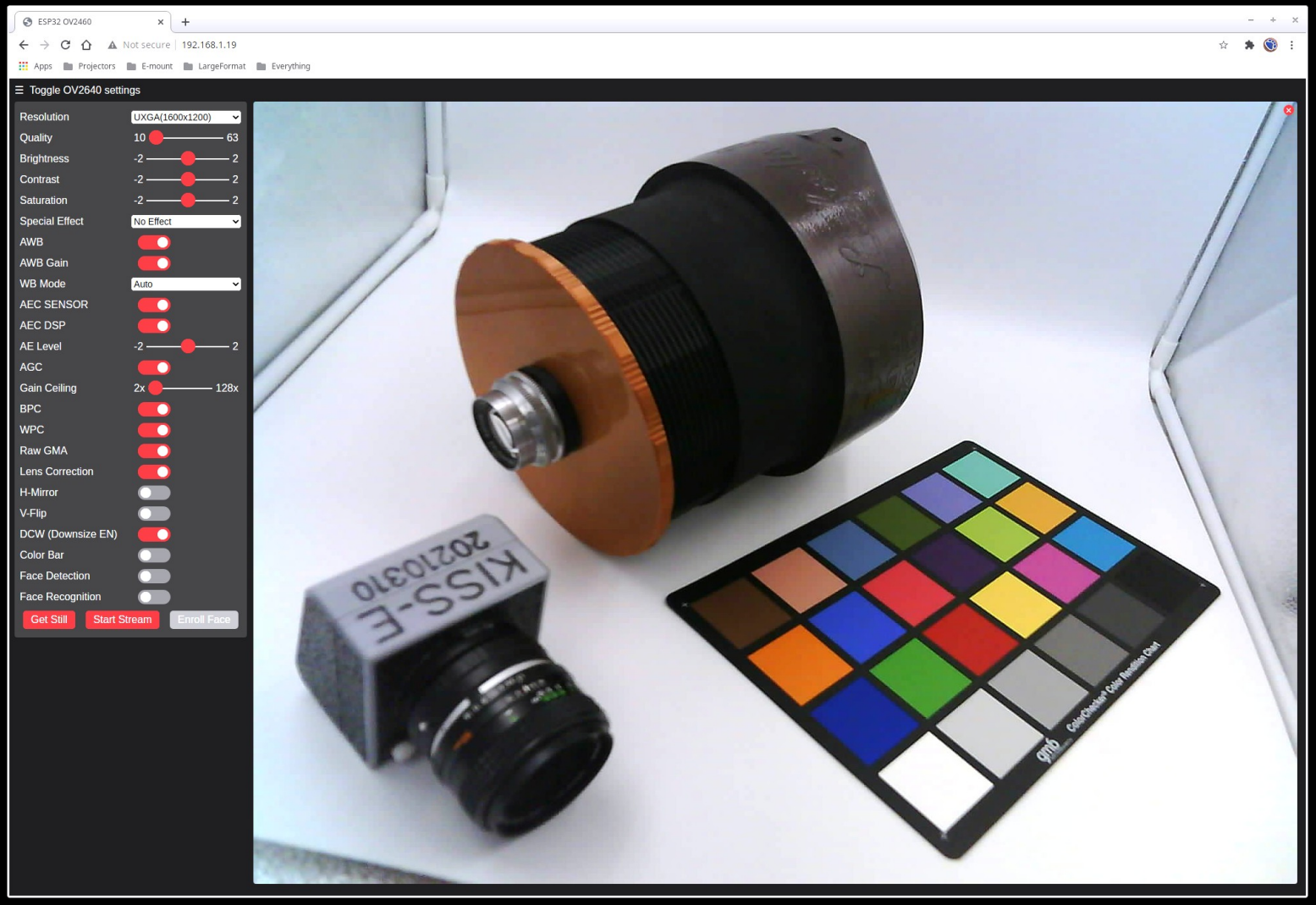
- **Open Memories** (2018-): targets Sony PlayMemories cameras, most discontinued
- Stanford **Frankencamera** (2010-14): API for Nokia N900... similar support in newer phones
- **Magic Lantern** (2009-): targets Canon EOS, no current models supported
- **CHDK** (2008-): Canon Hack Development Kit targets PowerShots/EOS, actively maintained

Potential Camera Platforms

Attribute or Feature	Mirrorless	Webcam	CHDK PowerShot	AI-Thinker ESP32-CAM
Image quality	$\geq$20MP,$\geq$12bpp	$\geq$0.3MP,$\geq$8bpp	20MP, 12bpp	2MP (1600$\times$1200), 10bpp
Exposure control	Extensive	Basic	Extensive	Basic plus some features
Interchange lens	Yes	Some models	No	Simple modification
Sensor size	$\geq$17.3$\times$13mm	$\geq$2.4$\times$1.8mm	$\sim$6$\times$4.5mm	3.59$\times$2.684mm
Near infrared (NIR)	Hard mod	Some models	Very hard mod	Simple mod
Wired connectivity	USB	USB	USB	UART, SPI, & I2C
Wireless	WiFi	No	Some models	WiFi & Bluetooth (with BLE)
Tethered control	Proprietary	Yes, UVC	Yes, CHDK PTP	Yes, programmable
Autonomous operation	Very limited	No	Yes	Yes
Programmable display	No	No	LCD	Options via connectivity
Programming support	No	No	CHDK C & Lua	Arduino & Espressif C/C++
Processor	Various ARM	?	Dual 80MHz ARM	Dual 240MHz Xtensa
Usable main memory	Varies	None	Several MB	520KB SRAM & 4MB PSRAM
Flash memory	SD card	No	SD card	4MB Flash & TF card
Power management	Minimal	No	Minimal	Modes from 310mA to 6μA
Sensor inputs	Camera UI	No	Camera UI	9 I/O pins; ADC, I2C, & SPI
Control outputs	Flash sync.	No	No	9 I/O pins; PWM, I2C, & SPI
Real time sync support	Remote	Some models	RTC, USB detect	RTC, programmable sync
Ease of embedding	Very hard	Moderate	Hard	Easy: 27x40.5x4.5mm board
Cost	$\geq$ \$500	\$8 – \$150	$\geq$\$100	$\sim$ \$7

University of Kentucky

ESP32-CAM Is An IoT Camera



University of Kentucky

What's Not-So-Great

- The **OV2640** camera
  - 15 FPS @ 1600x1200 is really ~6.2 FPS
  - Rolling shutter, limited exposure control
  - PSRAM too slow for full-res raw images
- The microcontroller packaging
  - MANY functions, **TOO FEW I/O pins**
  - No USB port, need USB-TTL adapter
  - Built-in antenna is easily blocked

University of Kentucky

Examples Using ESP32-CAM

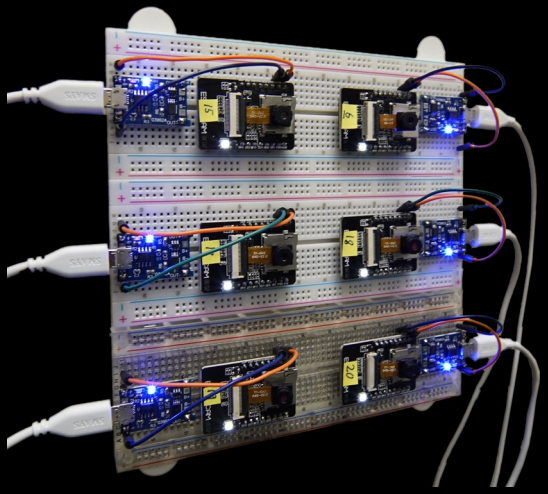
- **KameraflY**: compact rechargeable wireless remote-controlled swarm cameras (~$25)
- **KISS-E**: interchangeable-lens stand-alone camera with live view (~$25 + lens)
- **Lafodis**: scanning GP-class large-format camera (under $50 + lens)

All are fully programmable, use few electronics parts, and leverage 3D-printed mechanical parts

University of Kentucky

Camera Swarms

https://github.com/npsantini/KameraflY-ESP32-CAM-Camera-Swarm-System



- Camera swarm with WiFi control & sync
- ~$25 per stand-alone camera

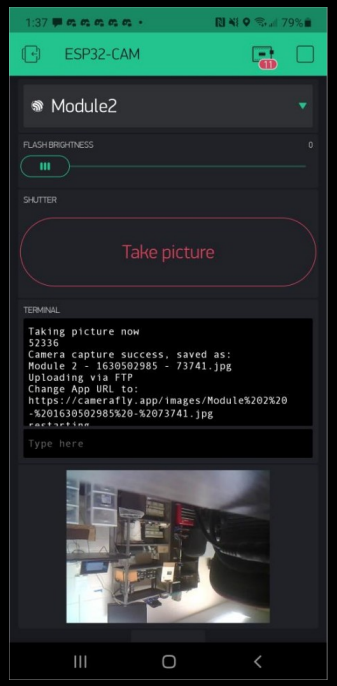
University of Kentucky

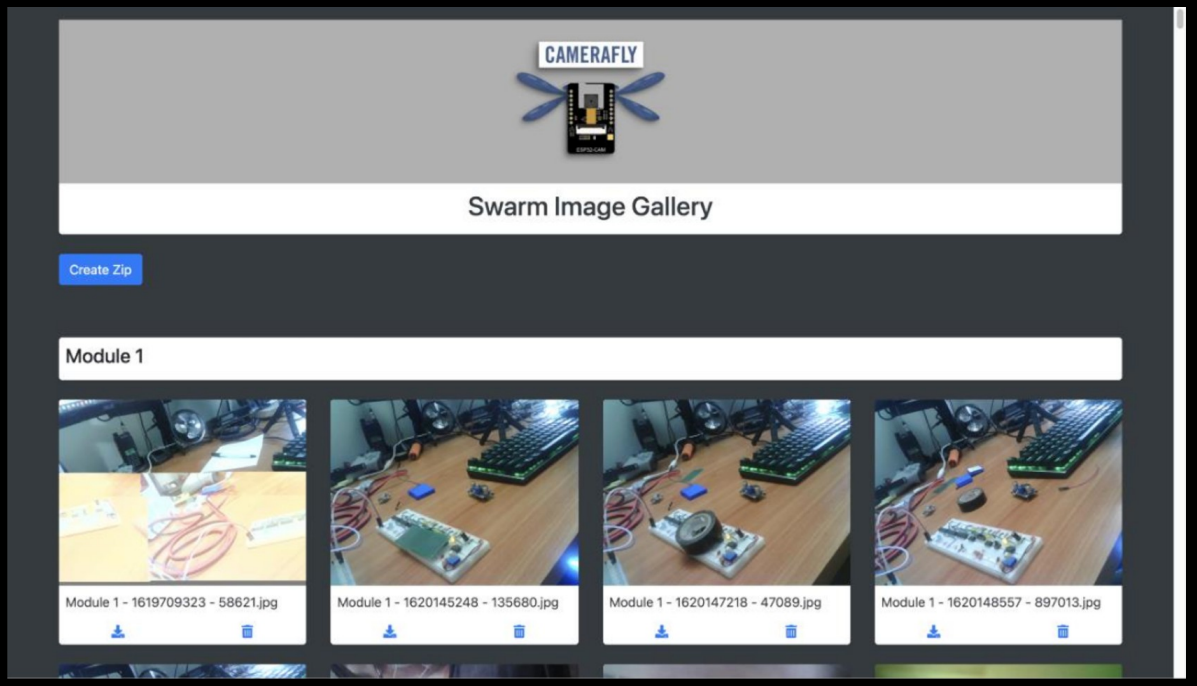
Hardware

- BP-70A LI battery, TP4056 USB charging
- Stand alone units, WiFi control

University of Kentucky

Software





- Blynk App for control with 1/100s sync.
- Web App swarm image gallery

University of Kentucky

Software

- Libraries used:
  - Blynk (Volodymyr Shymanskyy) v0.6.1
  - esp32_ftpclient (Leonardo Bispo) v0.1.4
  - SD (Arduino) v1.2.4
  - Time (Paul Stoffregen) v1.6.1
- Apache Web App swarm image gallery
  - Collects swarm images via FTP
  - PHP gallery and zip file packing

University of Kentucky

Kentucky's **I**nterchangeable-lens **S**mall **S**ensor – **E** mount

- **KISS-E** features
  - Stand alone camera with full programming and tether support
  - Use *any lens adapted to* **Sony E mount**, **~9.7x crop** compared to full frame
  - With/without NIR-blocking filter
  - Minimal OLED live display
  - Parts cost ~$25 + lens
- Also can be used with KameraflY software

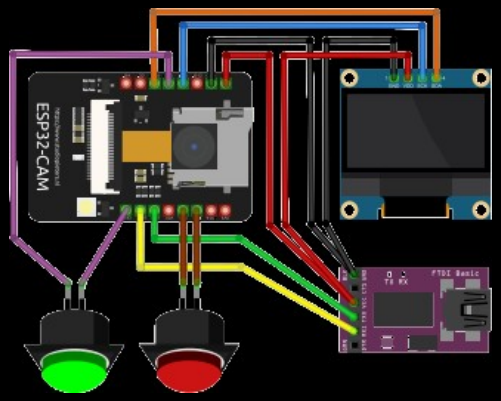
University of Kentucky

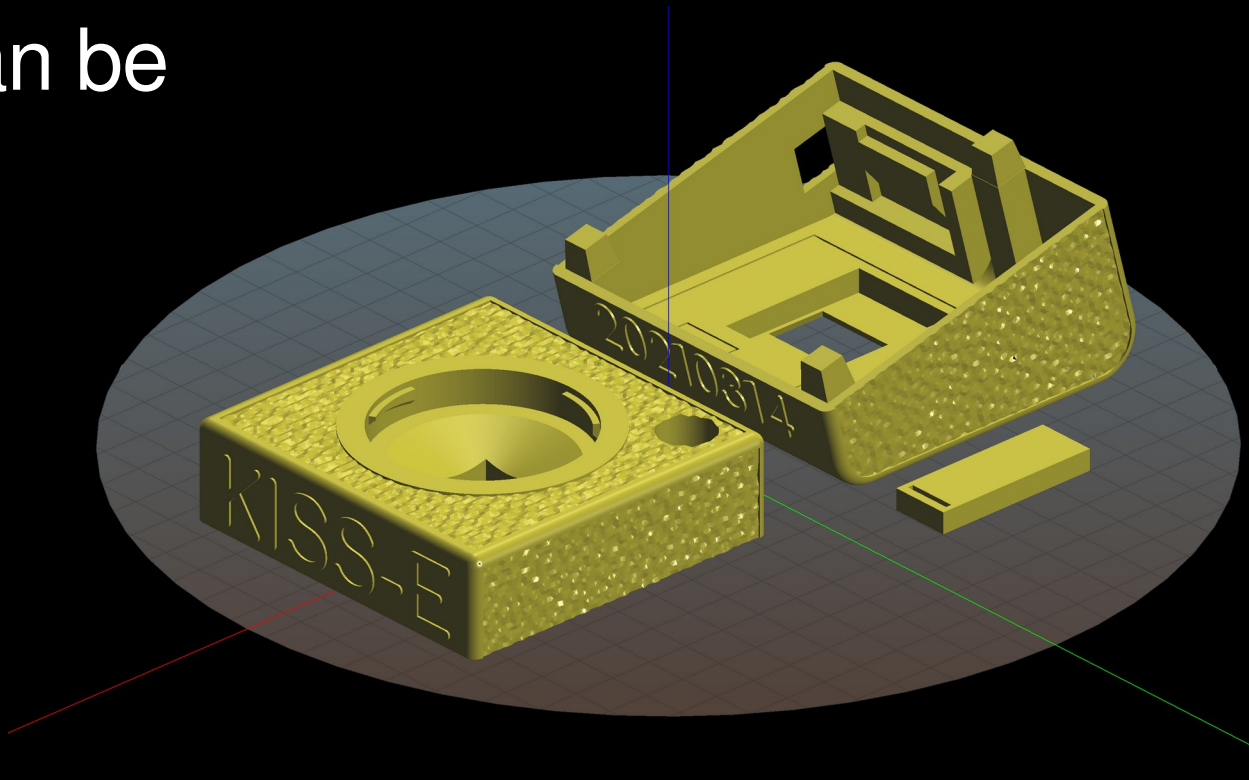
KISS-E (20210310 version)

University of Kentucky

KISS-E Parts

Connections can be wire wrapped





1 $7 ESP32-CAM
1 $3 SSD1306
1 $2.50 FT232RL (USB power)
2 $0.50 12mm Momentary SPST

~$1 PLA (~100g)

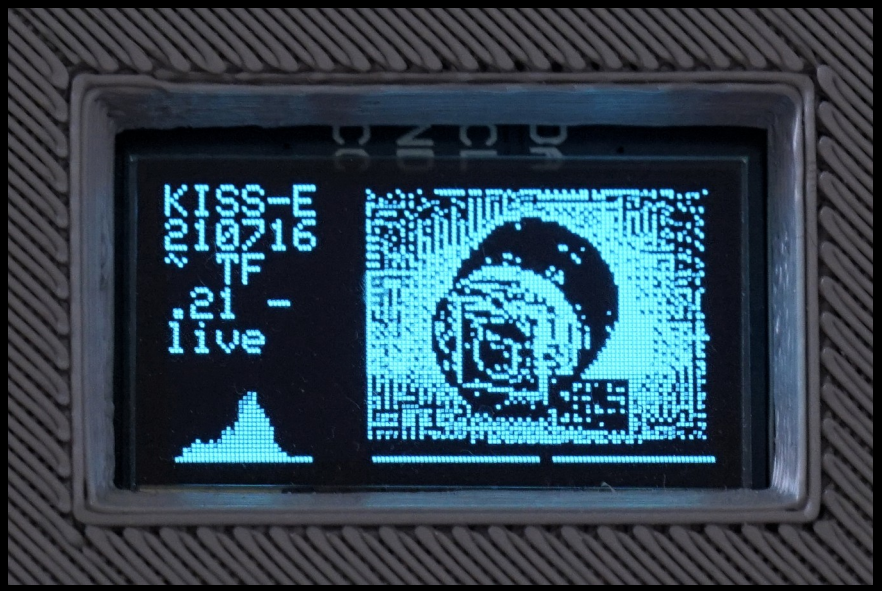
University of Kentucky

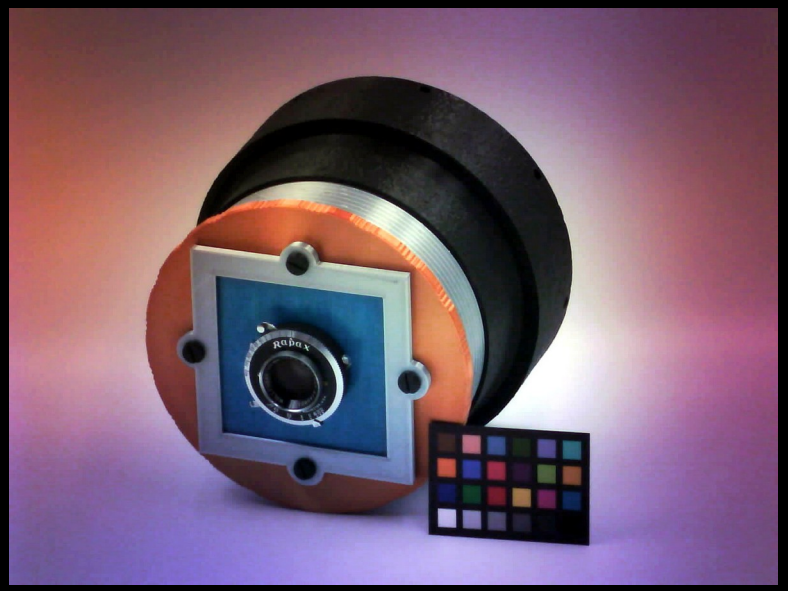
KISS-E Visible & Full Spectrum

- 2MP JPEGs with/without NIR filter
- Some off-axis pixel correction issues
- Slow electronic shutter

University of Kentucky

KISS-E Live Display





- 0.96” 128x64 OLED with SSD1306 driver
  - TF card % used, WiFi, capture state
  - Histogram and B&W rendered live view
  - Bar-gap focus aid: better focus, smaller gap

University of Kentucky

Large Format Digital Scanning: Lafodis

- 1st prototype was 4x5” X-Y scanning camera
- 2nd prototype, Lafodis160, at EI2021:

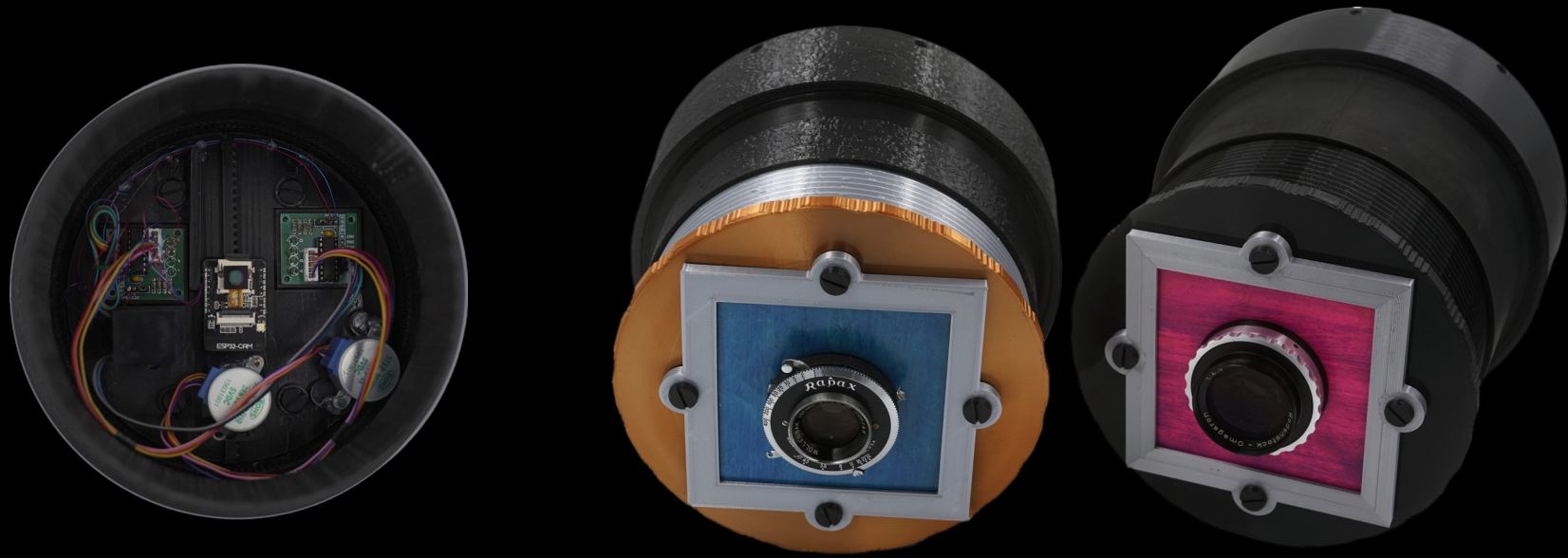
  An Ultra-Low-Cost Large-Format Wireless IoT Camera

  https://doi.org/10.2352/ISSN.2470-1173.2021.7.ISS-070

  - 160mm dia. radius-angle using steppers
  - Build cost <$50, or <$100 including lens
- 3rd & 4th prototypes of Lafodis:
  - Simplified construction, improved software
  - Arduino Pro Micro (via I2C) runs steppers
  - Herringbone gear drive

University of Kentucky

Lafodis, since EI2021



- WiFi interface, parameter tuning via browser
- 4x4” lens mount, 8x8mm (NIR) filter holder
- Still >2GP max resolution, new stepper library
- Scan order? See **P 14**, **ISS-199**

University of Kentucky

Conclusions

- IoT platform ⇒ easy to build custom cameras
  - Single-chip camera processor and sensor
  - Multi-core microcontroller, I/O, libraries
  - Easy to 3D print custom physical parts
  - **ESP32-CAM** is cheapest good option
- All our contributions *will be* linked from

  `http://aggregate.org/DIT/ESP32CAM/`

  - Sample research camera configurations
  - Our additions to the open source HW & SW

Aggregate.Org UNBRIDLED COMPUTING

University of Kentucky