**PRAGMA-EDGE Participants:**

Renato Figueiredo (UF) - renato@acis.ufl.edu
Kohei Ichikawa (NAIST) - ichikawa@is.naist.jp
Ken Subratie (UF) - kcratie@ufl.edu
Paul Hanson (UW) - pchanson@wisc.edu
Weicheng Huang (NCHC)
Sebastian [Osaka University - jsebastian@ais.cmc.osaka-u.ac.jp]
Martin Swany (IU) - swany@iu.edu
Aimee Stewart (KU)
Prapaporn (Nan) Ratanatamrong (Thammasat University) rattanat@gmail.com
Wassapon Watanakeesuntorn (NAIST) - wassapon.watanakeesuntorn.wq0@is.naist.jp
Vahid Daneshmand (UF) - vdaneshmand@acis.ufl.edu
Mai Nguyen (UC San Diego) - mhnguyen@ucsd.edu
Yutthana Boonpalit (nickname :Oat) (Thammasat University) - yutthana.boon@dome.tu.ac.th
Siwakorn Suwanjinda (nickname :Khing) (Thammasat University) -siwakorn.suw@dome.tu.ac.th
Kazuki Miyagoshi (Osaka University) - miyagoshi.kazuki@ais.cmc.osaka-u.ac.jp
Rob Quick (Indiana University)-rquick@iu.edu

**Raspberry Pi cluster - hardware:**
16 nodes at UF
8 nodes at NAIST

Action items:

- UF (Vahid) will provide documentation on how to add nodes (dedicated or advance notice if not available)
- Join Slack group for discussions
- Create a GitHub organization for software used in the project.
- Scheduling Zoom calls in early December and early February to track progress between now and PRAGMA-38

Potential additions:
Thammasat (Nan): 4 nodes (Raspberry Pi 3 and 4)

- Nan will also check with Kasetsart University as an additional site
- Check with NECTEC's vision processing group for possible applications (person recognition from videos on Intel NCS2)

Paul Hanson: might have 1-2 more nodes
IU (Martin): ? nodes
Weicheng Huang (NCHC): ? nodes

UF (Renato's team) and others interested - consider adding a few USB sticks (Intel Neural Compute Stick 2) for deep learning

**Raspberry Pi cluster - software:**

In addition to what we already have (Raspian, Docker, K8s, IPOP, autossh):
Istio - federated K8s

Additional functionality in containers:
Heron (reimplementation Storm)
Kafka
OpenWhisk
Telegraf - custom
EdgeFS
Deep learning libraries (e.g., TensorFlow, PyTorch, Keras, Caffe2)
Software emulation of a sensor streaming data (camera: VLC2VCam plugin, Martin - random time series generator)

**Applications of interest**
Nan: detecting people, events in CCTV streams (for public safety); possibly wildfires
Mai: wildfire detection (an existing project)
Paul: algal bloom detection in lake shores
Renato: license plate target detection/tracking
Kazuki: air conditioner/room temperature

**Open/research questions?**
Ken: Adaptive sampling? Sending sensors/drones?
Nan: smart city in Phuket (?) - camera movement, altering configuration

**Miscellaneous discussion items:**

Mai
Wildfire: cameras - images to detect motion/smoke
HPWREN sensors
Analyze to detect Santanna conditions (can lead to fire)
Analyze satellite imagery

Hammer traps

off-the-shelf deep learning

need good training data

under different conditions?

need to deploy cameras and gather data for a season to build data sets

Application: Detection and/or prediction of harmful algal blooms; specifically, surface scum-forming algal blooms. Madison lakes frequently have such blooms, especially when there has been a wet spring preceeding the hot summer months. Surface scums are patchy and dynamic and dangerous for public health. Video would be one way to monitor for surface scums.

Paul: 1 picture per minute should be enough. Mounting video on top of the Center for Limnology for this purpose could be a good test case.  Mendota and Menona; half a dozen beaches: humans on the ground making assessment (lifeguards, county/city responding to a lifeguard, take water sample)

**Reverse SSH Tunneling to Connect to RPi Cluster Master Node:**
https://github.com/ipop-project/ipop-project.github.io/wiki/Reverse-SSH-Tunneling-to-UF-Server

**RPi Cluster Slack Channel:**
https://rpi-cluster.slack.com