GUEST ALARM SYSTEM APPLIED IN SMART HOME

Tran Quoc Thanh – CH2001035
Nguyen Thanh An -
Vo Be Ngoan -

Tran Quoc Thanh
CH2001035

Content

- Objective
- System design
- Compare and select components (camera, IoT node hardware, algorithm)
- Experiment result
- Summary

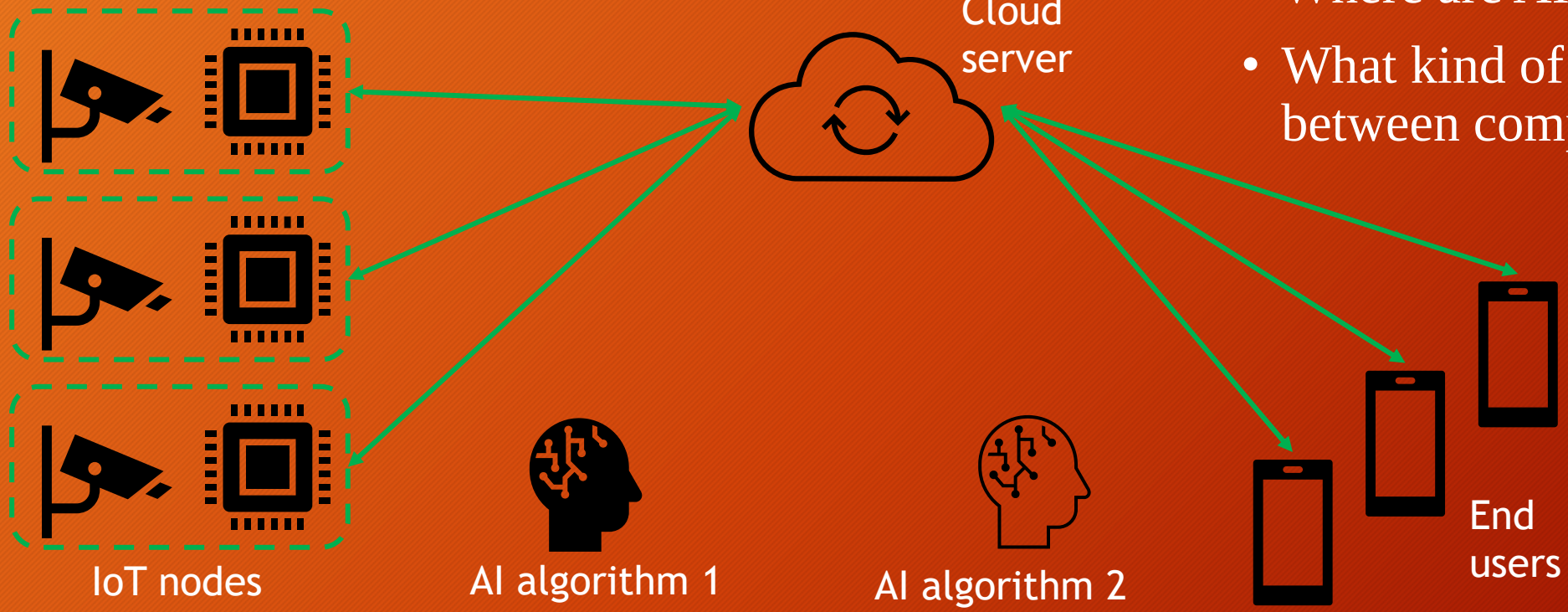
Objective

- Design system architect for guest alarm system applied in smart home
- Compare and choose hardware for IoT node and face recognition algorithm
- Experiment: compare hardware and algorithms in term of speed and accuracy with different conditions

System design

Trần Quốc Thành

System’s components



- Where are AI algorithms stored?
- What kind of data is transferred between components

Option 1: AI algorithms on cloud

Advantage

- Utilize cloud's computational power
- Low technical requirements lead to low cost of IoT node
- Utilize big data collected from users

Disadvantage

- Require large network bandwidth
- System stop working when network unavailable
- Required cloud's computational power and storage power increase rapidly with number of IoT nodes

Solution 2: AI algorithms on IoT node

Advantage

- Minimal amount of network bandwidth required
- Local IoT node (and connected actuator if any) can still work without network connection
- User’s privacy is secured

Disadvantage

- High technical requirements lead to high cost of IoT node
- Limited AI algorithms due to IoT node’s computational and storage capability
- Challenging to ensure security

Solution 3: Hybrid (face detection on IoT node, face embedding and recognition on cloud)

Advantage

- Minimal network bandwidth is required
- Sophisticated face embedding algorithm can be used
- Better security (important model stays on cloud)
- Reduced requirements on cloud's computational and storage

Disadvantage

- System stop working when network is unavailable

System design following solution 2

IoT node hardware and algorithm

Trần Quốc Thành

Camera

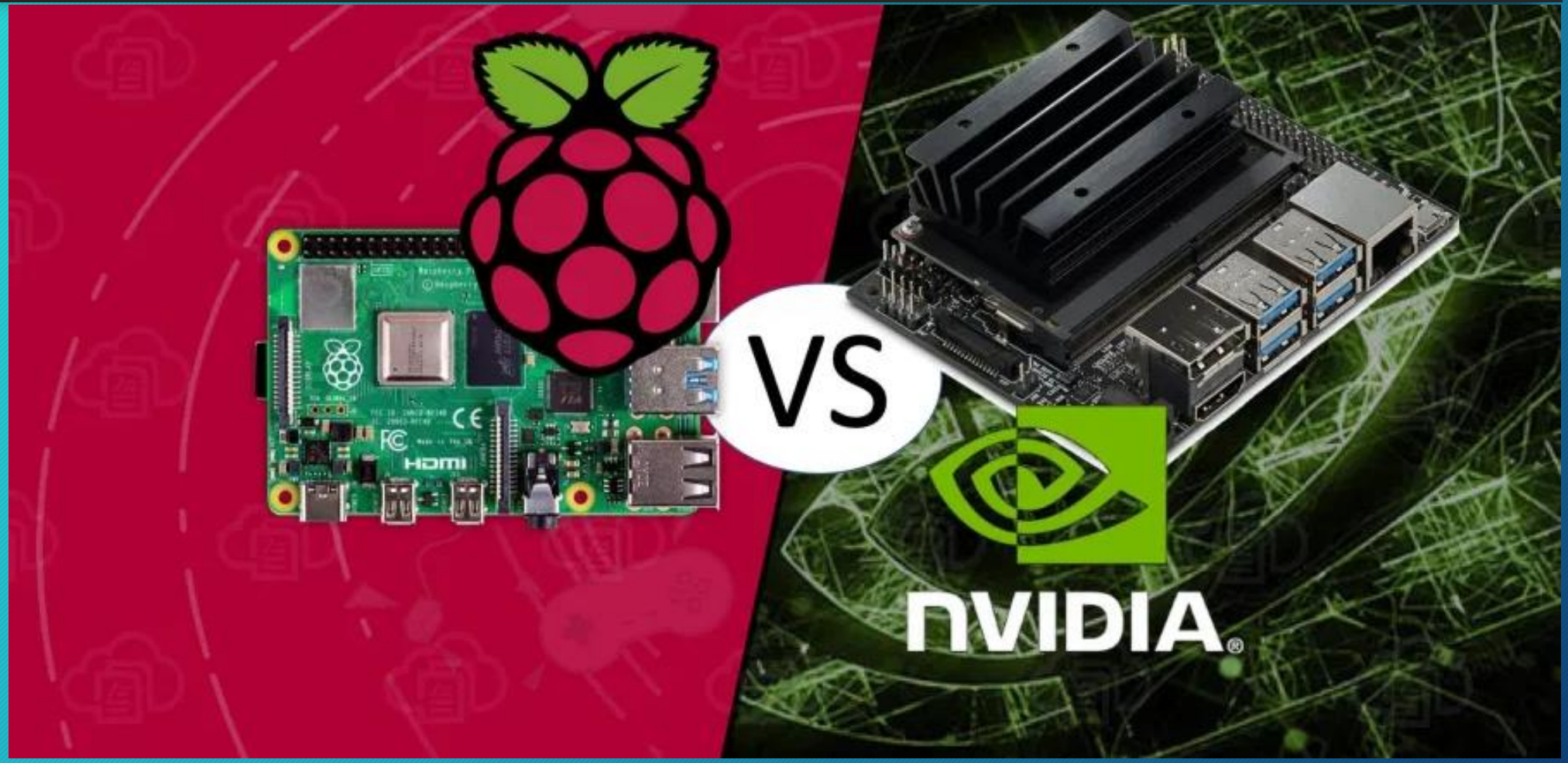
Chi tiết	Pi camera V2	HQ camera	Chosen camera
Size	25x24x9mm	38x38x18.4mm	30x30x40mm
Still resolution	8Megapixels	12.3Megapixels	?
Video modes	1080p30 720p60	1080p30 720p60	512p30
Optical size	1/4,0”	1/4,0”	1/2,7”
Focal length	3.04mm	Depends on lens	3.6mm
Bit depth	10	12	?
Price	25$	50$	25$

Camera

Distance: 0.5m

Distance: 4m

Raspberry Pi 4 vs Nano Jetson

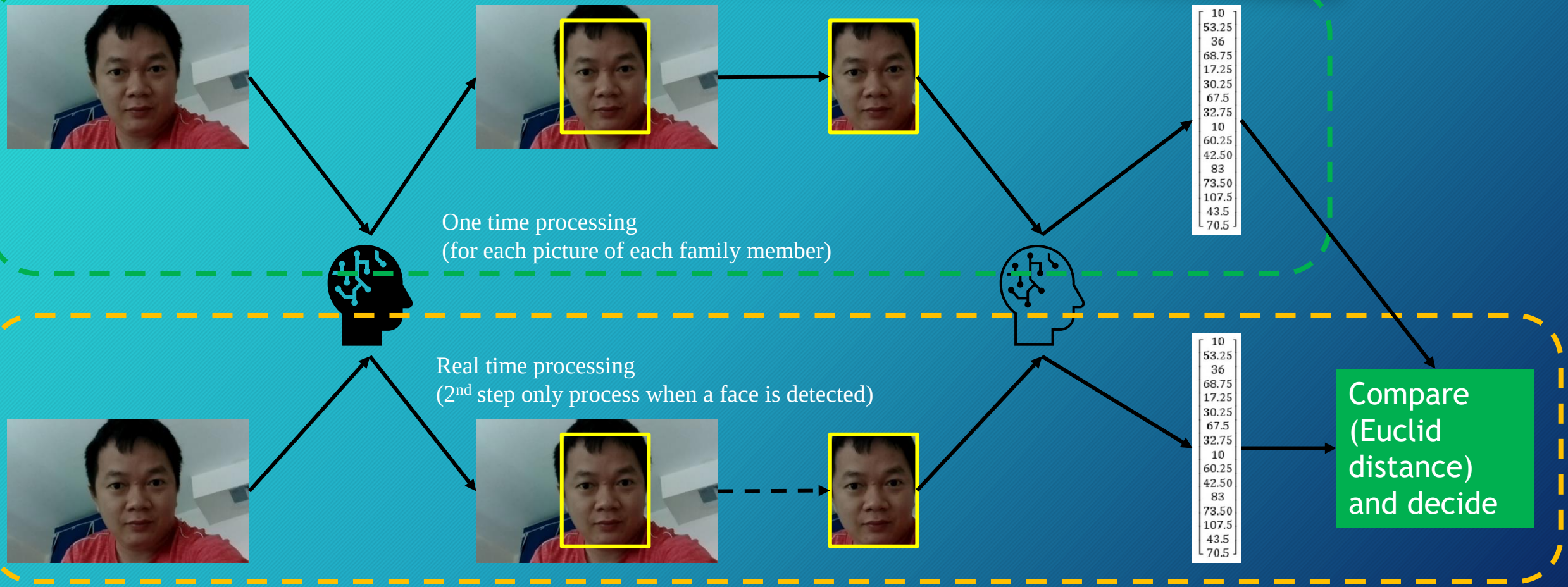


This picture and table below based on information from: Raspberry Pi 4 Vs NVIDIA Jetson Nano Developer Kit | Build5Nines

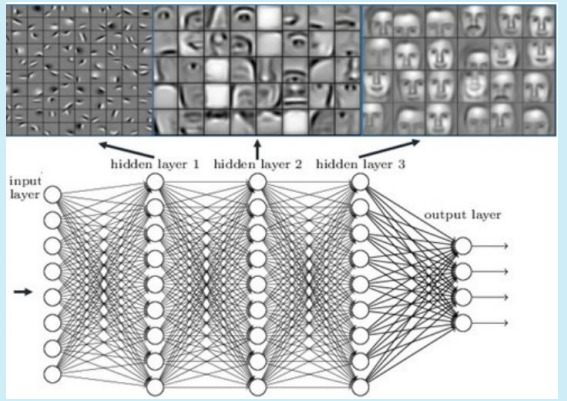
Raspberry Pi 4 vs Nano Jetson

Item	Raspberry Pi 4	Nano Jetson
CPU	Quad-core ARM Cortex-A72 64-bit @ 1.5 Ghz	Quad-Core ARM Cortex-A57 64-bit @ 1.42 Ghz
GPU	Broadcom VideoCore VI (32-bit)	**NVIDIA Maxwell w/ 128 CUDA cores @ 921 Mhz**
Memory	4 GB LPDDR4	4 GB LPDDR4
Networking	Gigabit Ethernet / **Wifi 802.11ac**	Gigabit Ethernet / M.2 Key E *(for Wifi support)*
Display	2x micro-HDMI *(up to 4Kp60)*	HDMI 2.0 and eDP 1.4
USB	2x USB 3.0, 2x USB 2.0	4x USB 3.0, USB 2.0 Micro-B
Other	40-pin GPIO	40-pin GPIO
Video Encode	H264(1080p30)	H.264/H.265 (4Kp30)
Video Decode	H.265(4Kp60), H.264(1080p60)	H.264/H.265 (4Kp60, 2x 4Kp30)
Camera	MIPI CSI port	MIPI CSI port
Storage	Micro-SD	Micro-SD
Price	**$55 USD**	$99 USD

AI algorithm



AI Algorithm – face detection

Item	Haar-cascade	Histogram of Oriented Gradient (HOG)	CNN
			
Idea	Look for simple pattern through image on different scale	Use oriented gradient (white arrow) to look for pattern through image on different scale	Hierarchy of abstract features are learnt from deep network. CNN ensure faces with different position/scale on image are detected
Model size	Very small	Small	Big
Speed	Very fast	Fast	Slow (can be speed up by VGU)
Quality	Acceptable	State of the art	Highest

The information in this table is based on Microsoft Word - Nicolas Delbiaggio - 06.05.17 with template.docx (core.ac.uk)

AI Algorithm – face embedding

Item	Eigenfaces	Fisherfaces	CNN
Idea	Using PCA (principle component analysis) to compress high dimensional image into below 100-dim vector	Also use PCA but consider also the average of each class (each person) using Linear Discriminant Analysis for better result	A pretrained neural network on faces can be used for face embedding in forward propagation. The output from 2nd last fully connected layer is the needed embedding vector
Model size	No model needed	No model needed	Big
Speed	Very fast	Very fast	Slow (can be speed up by VGU)
Quality	Acceptable	Acceptable	Very good

Use RESNET model from dlib library (RESNET-29): link
Dataset include 3 million faces of about 7,500 individual worldwide.

The information in this slide is based on Microsoft Word - Nicolas Delbiaggio - 06.05.17 with template.docx (core.ac.uk) and A Study of Face Embedding in Face Recognition (calpoly.edu)

Server and mobile app

Nguyen Thanh An

Supporting content

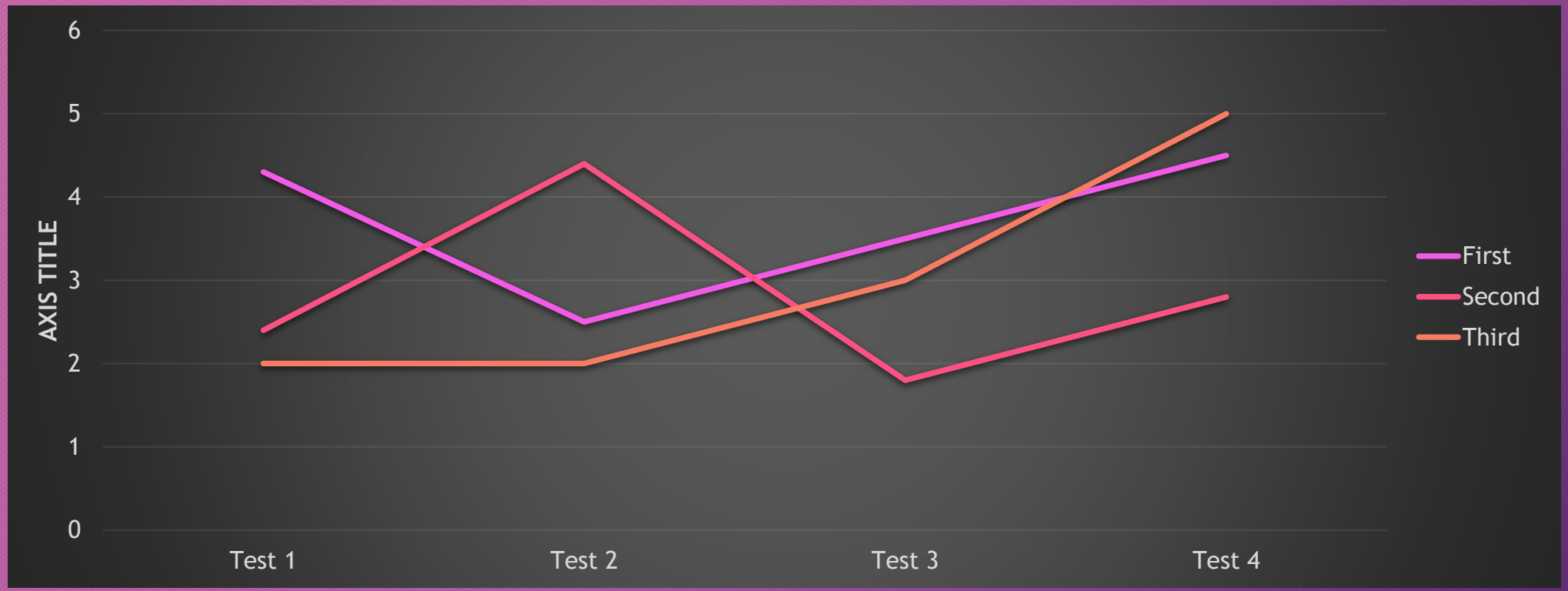
Heading

- List item
- List item
- List item

Heading

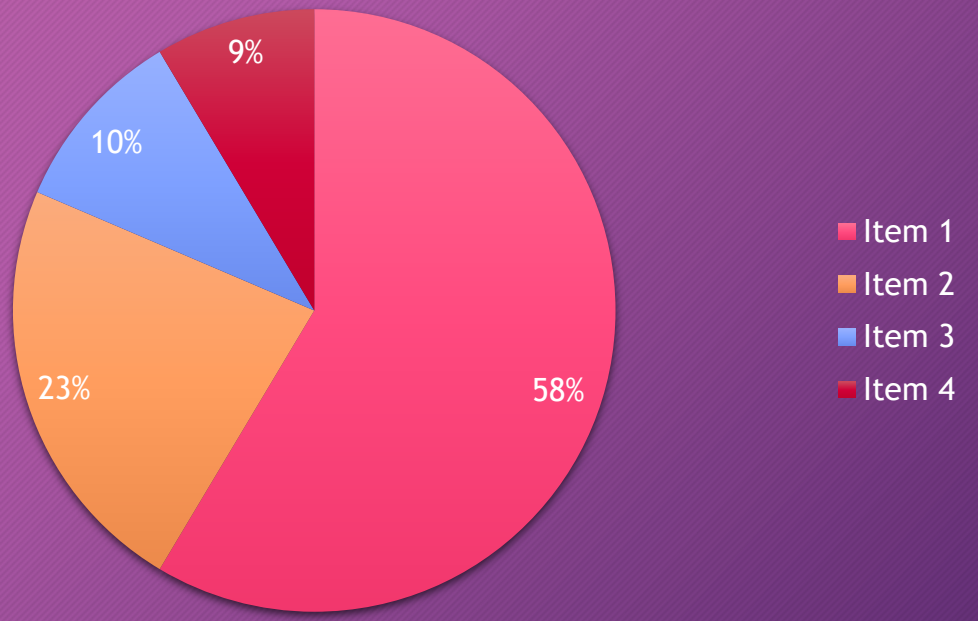
- List item
- List item
- List item

Data observations



Data Observations

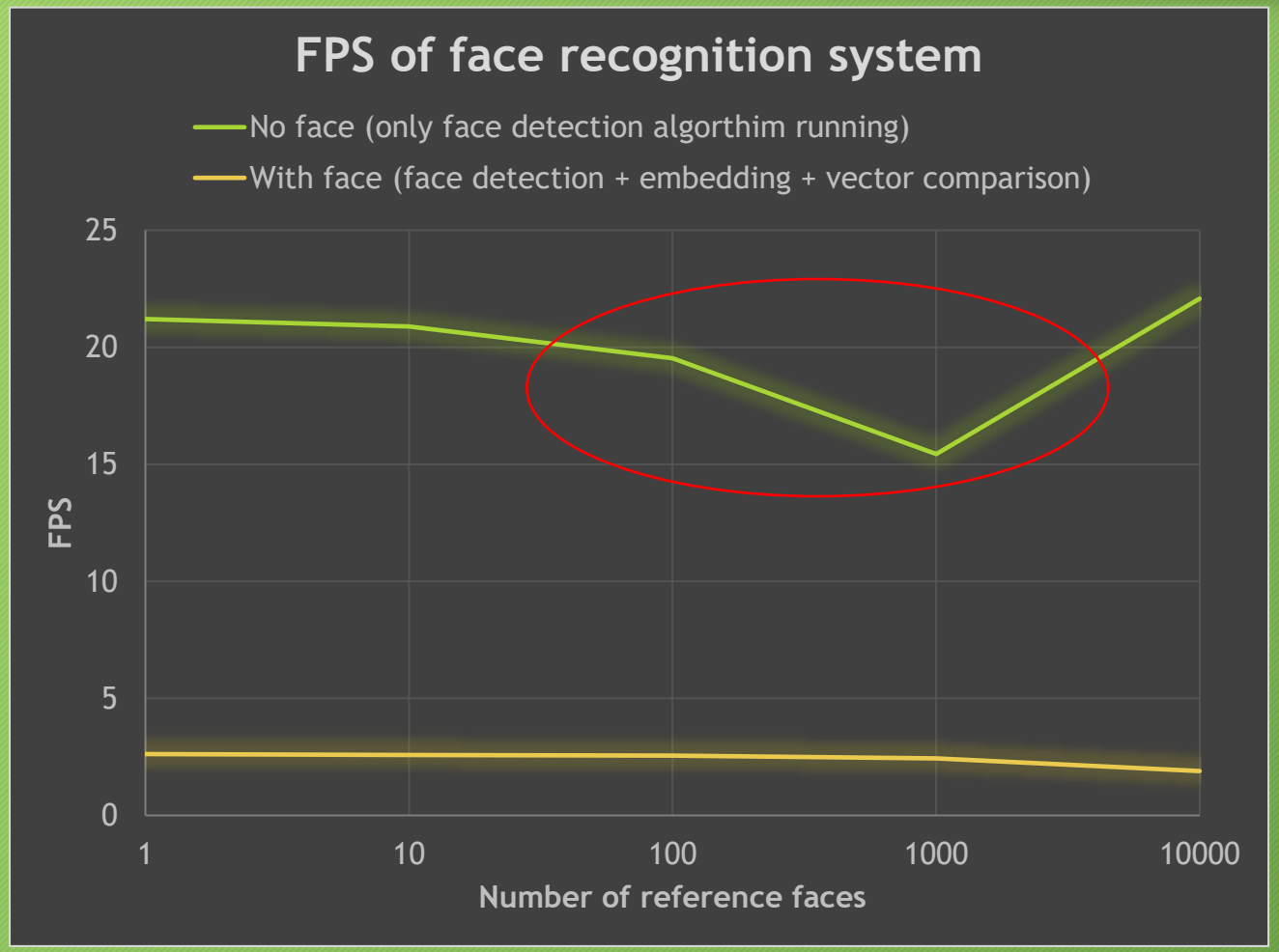
Optional statement



Experiment result

Tran Quoc Thanh

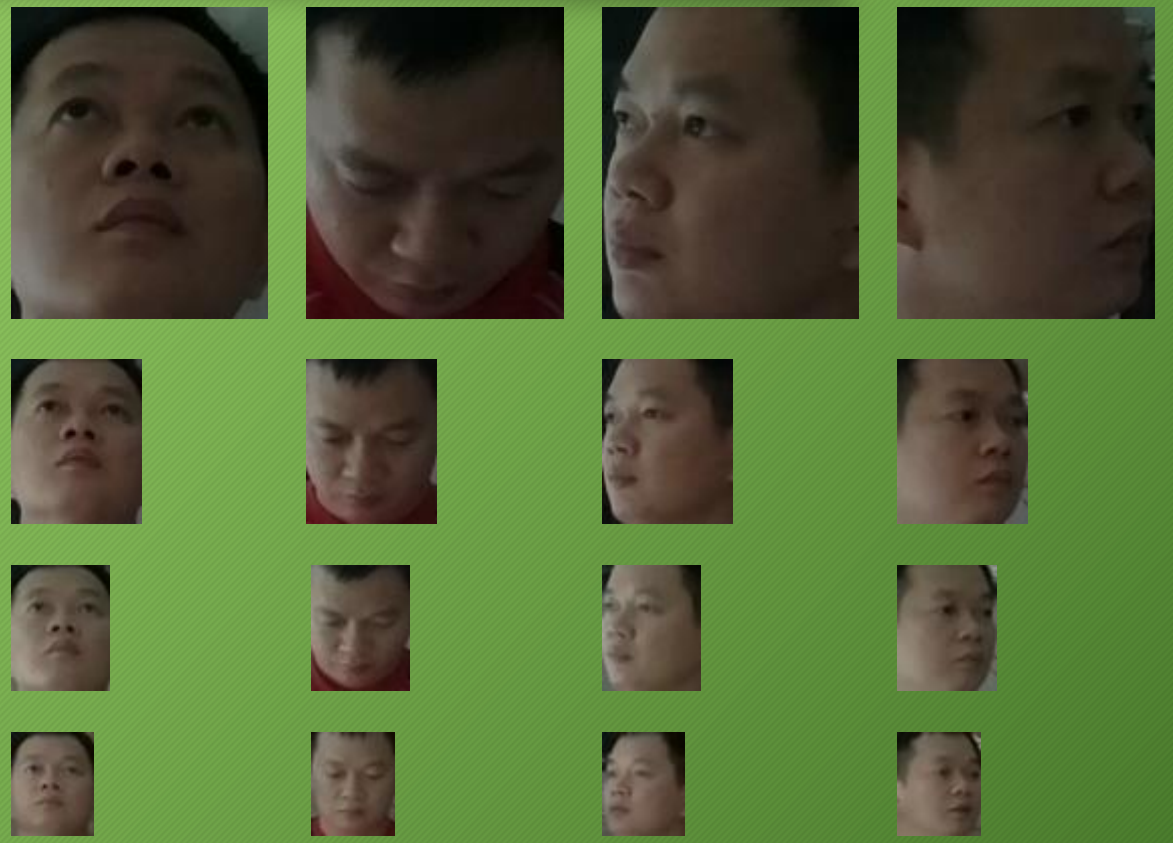
Experiment 1: Speed



- Face detection algorithm (HOG) run very fast
- Face detection algorithm should be independent of number of reference faces (of family members)
- Face embedding algorithm (RESNET-29) run very slow (FPS reduced from ~22 to 2.6). IoT node with GPU can improve performance.
- Time to compare guest face to references face doesn't matter up to 1,000 reference faces. Above that, it starts to affect performance (2.5 FPS -> 1.9FPS)

Experiment 2 – face detection ability

Distance [m]	Up [degree]	Down [Degree]	Left [Degree]	Right [Degree]
0.6	60	45	70	70
1.2	60	45	65	65
1.8	45	45	45	45
2.1	30	30	35	35



- At 0.6m, the range of angle so that face is detected is quite big
- At 2.1m, the range of angle so that face is detected is reduced to around 30degree.
- Within these ranges, faces are detected 100% with different emotion.
- Outside these range, detectability significantly reduced

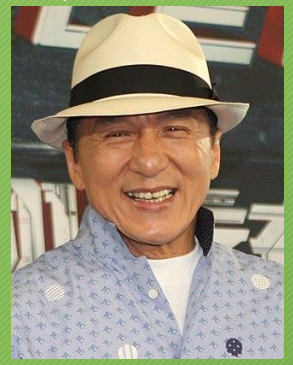
Experiment 2 – face detection ability (continue)

- Because of camera quality, with dimmer light (early in the morning or later afternoon), picture from camera is black.
- Even with better light and camera can capture reasonable picture, system can’t detect face.

Experiment 3 – recognition accuracy

- The test is done for only 10 people below
- With only one reference picture/person, the accuracy is about 50%
- With 2 pictures/person, the accuracy is 100% (with all distances and angles that a face is detected)










Summary

Trần Quốc Thành

Summary

✓ Done

- Study, design and build a guest alarm system applied in smart home including IoT node, server and mobile app.
- Conduct experiment to find the performance of built system (in term of speed, face detection and face recognition capability)

❖ Future work

- Conduct experiment with Jetson Nano and compare with Raspberry Pi 4
- Conduct experiment with other face embedding algorithms and compare it with RESNET-29
- Quantify lighting condition
- More accuracy angle measurement

❑ What can we do better:

- Team work (forming–storming–norming–performing)