Thai sign language recognition from video
การรู้จำภาษามือของไทยจากวีดีโอ

1. Kitithat Pansang
2. Pongsarat Chootai
3. Theethut Narksenee
4. Nidchapan Nitisukanan
5. Saranchai Angkawinijwong

Introduction

- The sign language translation from an image or sequential data format in the video is appropriate to apply deep learning subjects together with image processing knowledge to help solve problems. Since we found that the previous research on sign language translation related to Thai is minimal, the researchers were interested in designing the subject to translate Thai sign language using data from video image analysis. This study focus on Sign language recognition (SLR) which is a translation of words that the communicator wants to communicate from a video that shows the gestures of the communicator, which is a basic sign language translation and will also be able to extend to more detailed tasks in the future, for example, continuous sign language recognition (CSLR) and sign language translation (SLT).

Scope of work

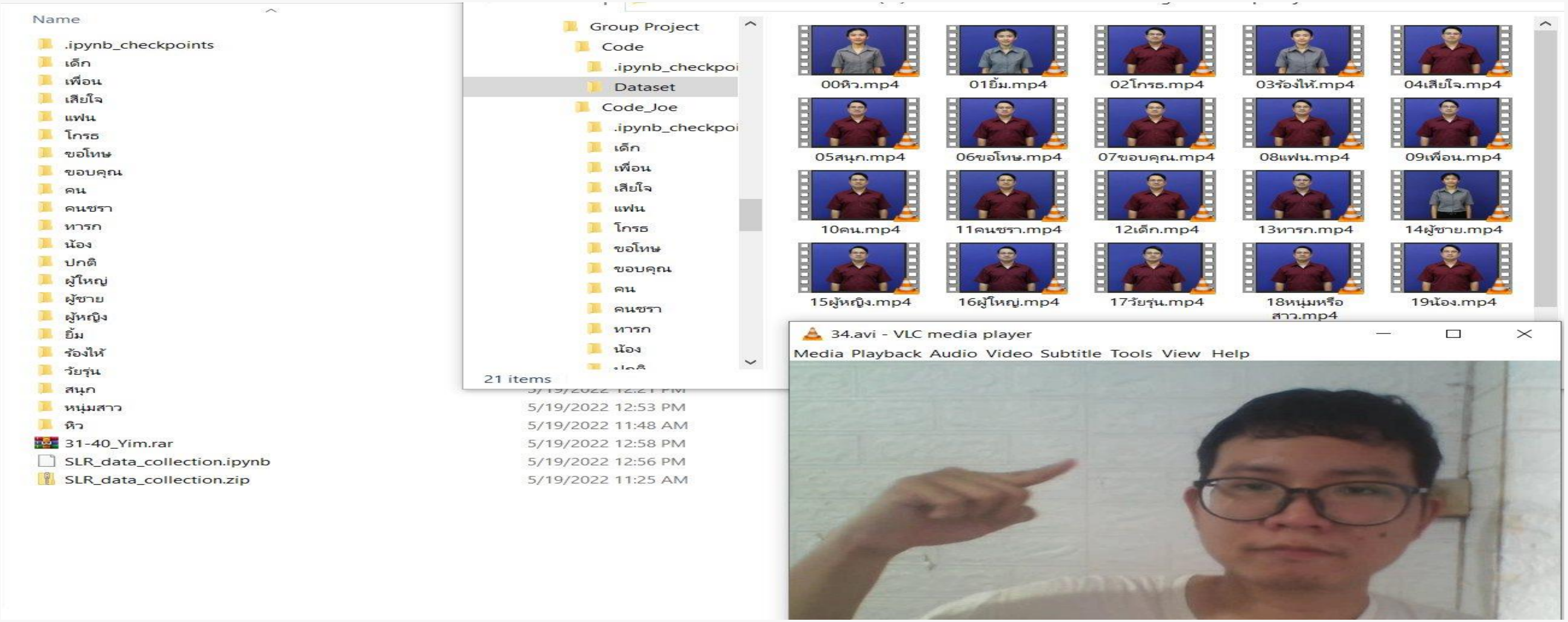
- This study would focus on Sign language recognition (SLR) using the created 20 gestures of sign language dataset, which consists of Child(เด็ก), friend(เพื่อน), sad(เศร้า), lover(คนรัก), angry (โกรธ), sorry(ขอโทษ), thanks(ขอบคุณ), person(คน), old person(คนแก่), infant(ทารก), brother/sister(พี่น้อง), adult(ผู้ใหญ่), men(ผู้ชาย), women(ผู้หญิง), smile(ยิ้ม), cry(ร้องไห้), adolescence(วัยรุ่น), fun (สนุก), youthful(หนุ่มสาว), hungry(หิว).
- Therefrom, apply the Deep learning models namely, LSTM, RNN, and GRU to compare each model's performance by a confusion matrix.

Expected benefit

- To enhance the ability for deaf people to communicate with normal people by understanding the meaning of the words from sign language through motion performance in real-time.

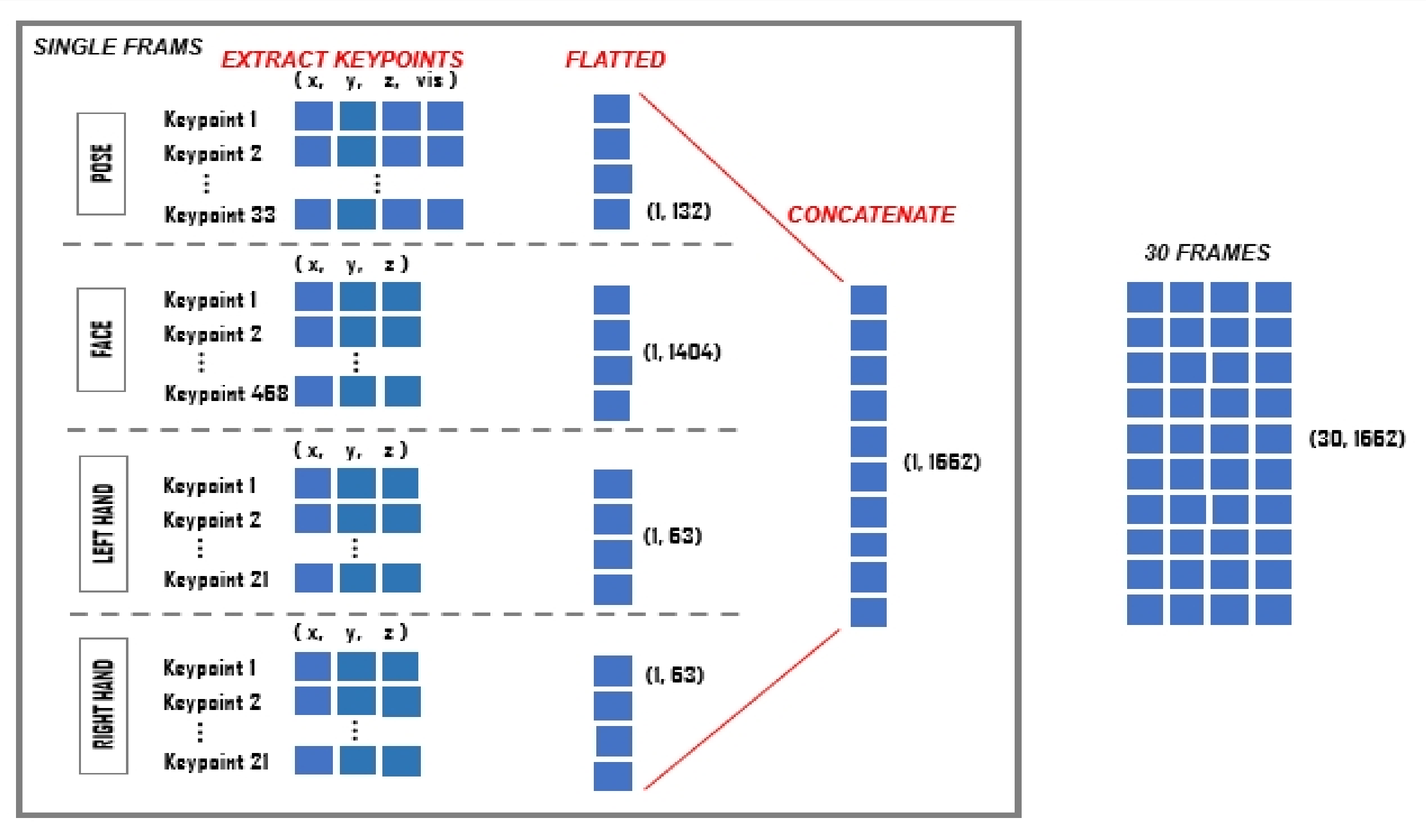
COLLECTING DATA

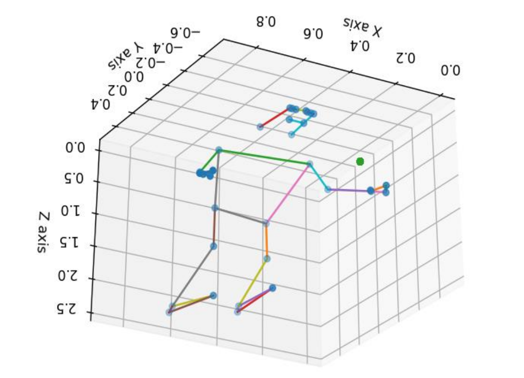
Firstly, we'll try learning Thai sign language from National Association of the Deaf in Thailand (NADT). Then, we record the short video clip, 3 seconds with 30 fps each, for 20 + 1 selected Thai sign language words.



COLLECTING DATA

We then apply the media pipe to extract the key points represent the action of hands, arms , and face. The total 1,050 arrays collected location of each key point will be accumulated and finally be used for deep learning models training.

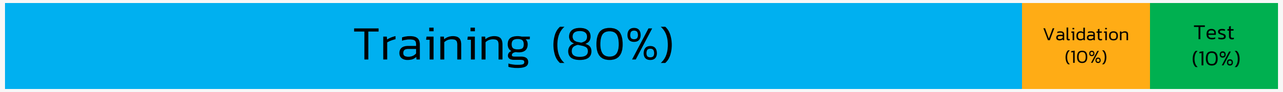




Model experiment

The total 1,050 data will be split into training , validation, and test set in portion 80 : 10 : 10 accordingly. Then, it'll be used for training 3 deep learning models:

1. LSTM
2. RNN
3. GRU



Epoch = 200, Batch Size = 32, Random State = 42

The generated model will be evaluated with the following details optimizer='Adam', loss='categorical crossentropy', metrics=['categorical accuracy'])

Model experiment

1. LSTM

Model: "sequential_4"

Layer (type)	Output Shape	Param #
lstm_11 (LSTM)	(None, 30, 32)	216960
lstm_12 (LSTM)	(None, 64)	24832
dense_12 (Dense)	(None, 64)	4160
dense_13 (Dense)	(None, 32)	2080
dense_14 (Dense)	(None, 21)	693

Total params: 248,725
Trainable params: 248,725
Non-trainable params: 0

2.RNN

Model: "sequential_19"

Layer (type)	Output Shape	Param #
simple_rnn_33 (SimpleRNN)	(None, 30, 128)	229248
simple_rnn_34 (SimpleRNN)	(None, 256)	98560
dense_39 (Dense)	(None, 128)	32896
dense_40 (Dense)	(None, 64)	8256
dense_41 (Dense)	(None, 21)	1365

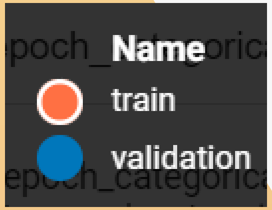
Total params: 370,325
Trainable params: 370,325
Non-trainable params: 0

3.GRU

Model: "sequential"

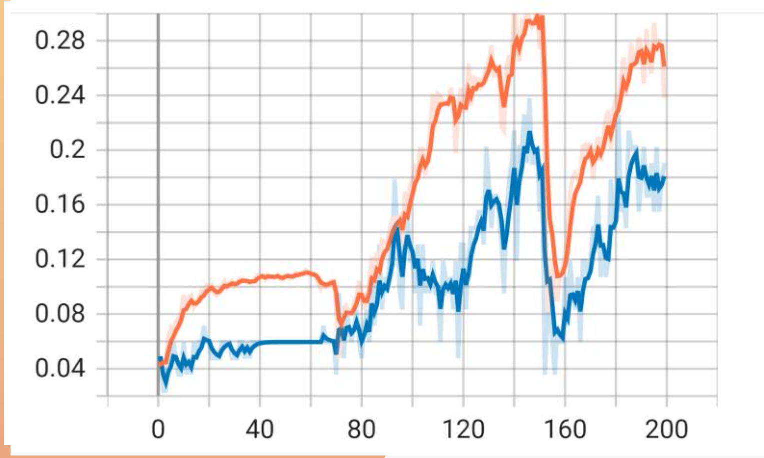
Layer (type)	Output Shape		Param #
gru (GRU)	(None, 30, 128)	('relu')	688128
gru_1 (GRU)	(None, 256)	('relu')	296448
dense (Dense)	(None, 256)	(None)	65792
dense_1 (Dense)	(None, 128)	(None)	32896
dense_2 (Dense)	(None, 21)	('softmax')	2709

Total params: 1,085,973
Trainable params: 1,085,973
Non-trainable params: 0

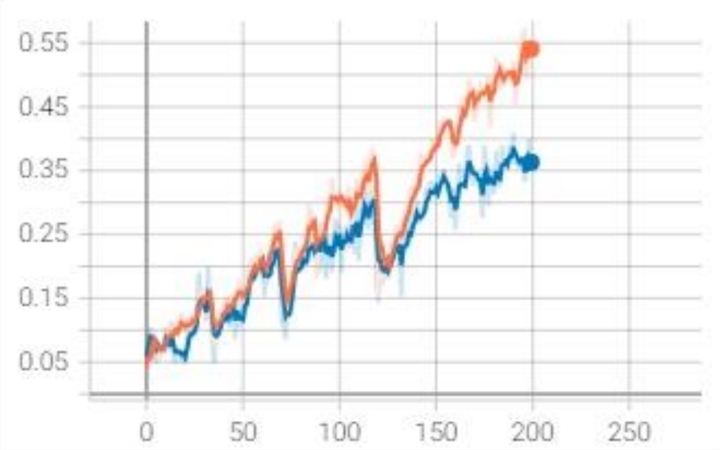


Result Comparison

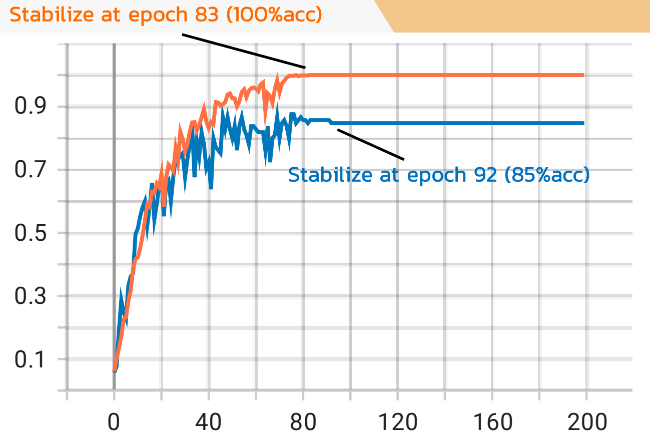
LSTM ACCURACY



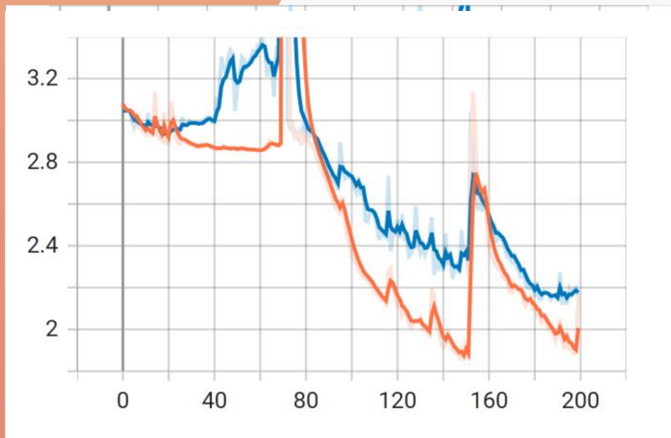
RNN ACCURACY



GRU ACCURACY

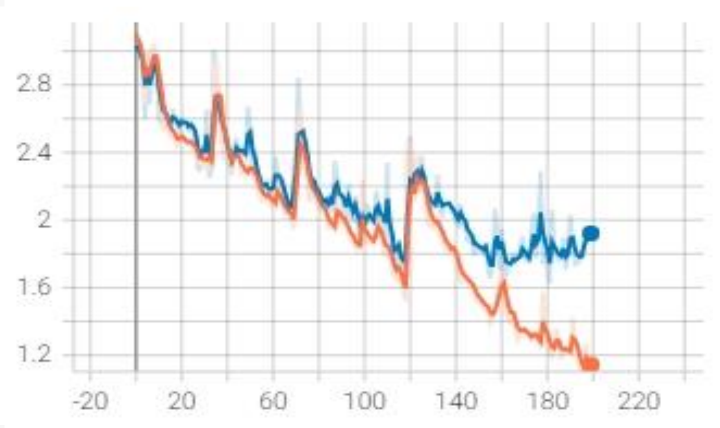


LSTM LOSS



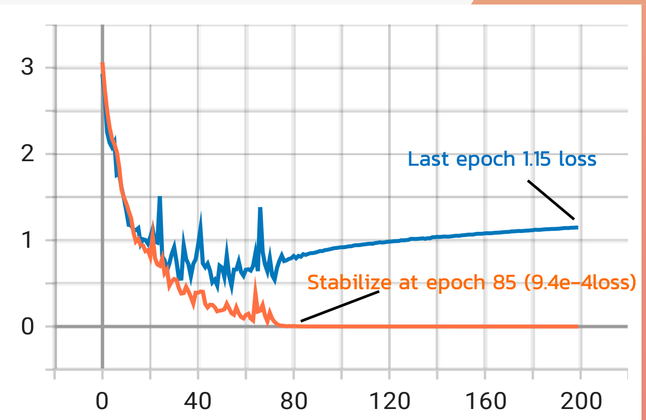
Test set acc: 29.7%

RNN LOSS



Test set acc: 34.3%

GRU LOSS



Test set acc: 84%

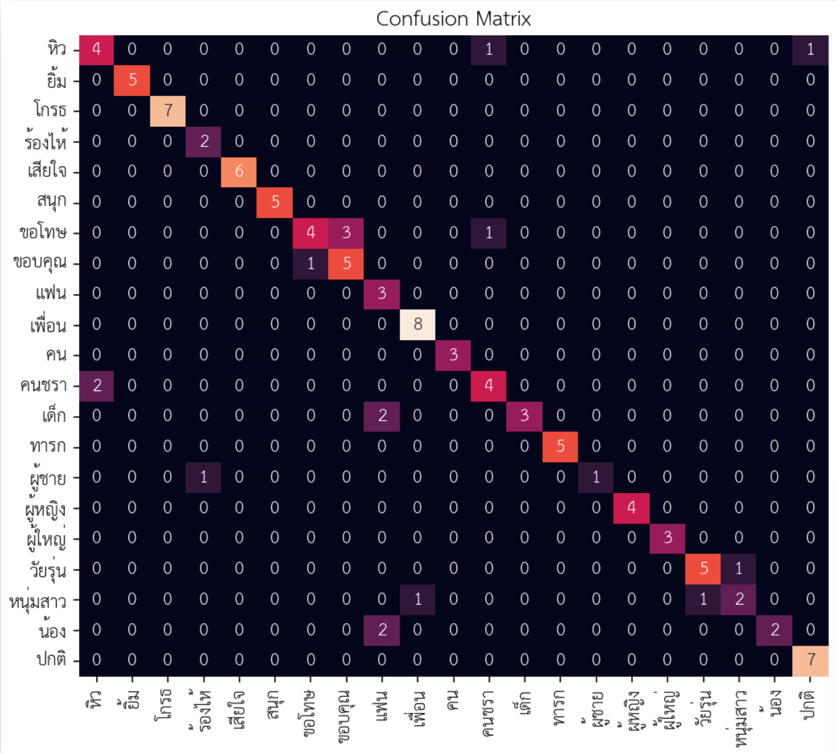
Best Model (GRU) Confusion Matrix (Test set)

-Most prediction looks fine / the **test set accuracy** is **84%**

-There is some confusion between predicting class "**ขอโทษ**" and "**ขอบคุณ**" since both classes have only a **facial posture** difference

-**Lowest precision** class is "**แฟน**" (0.43) due to a **similar initial hand movement** to class "**เด็ก**" and "**น้อง**"

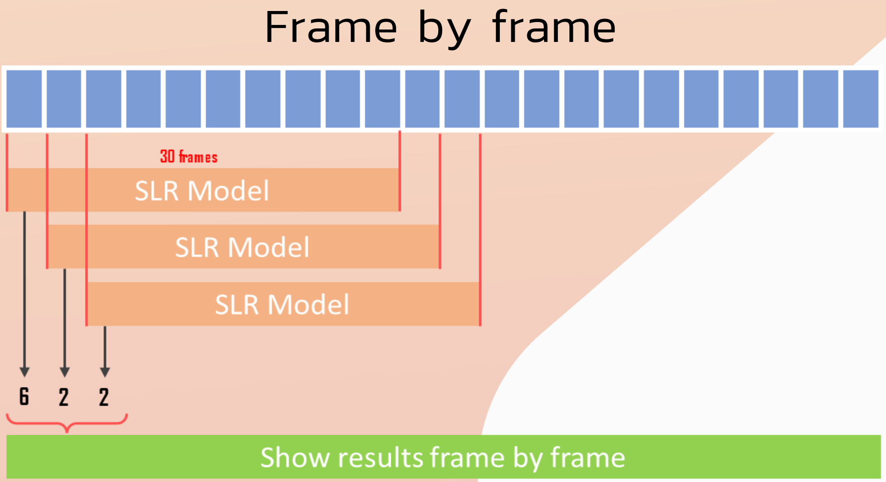
-**Lowest recall** classes are "**ขอโทษ**", "**ผู้ชาย**", "**หนุ่มสาว**" and "**น้อง**" (0.5) because these classes have a **similar posture** to some other classes

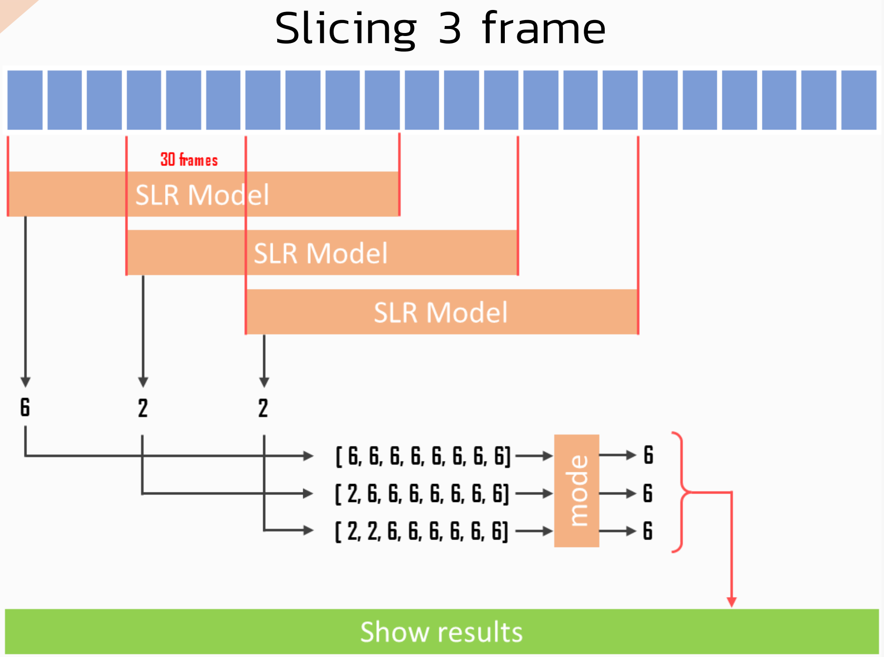


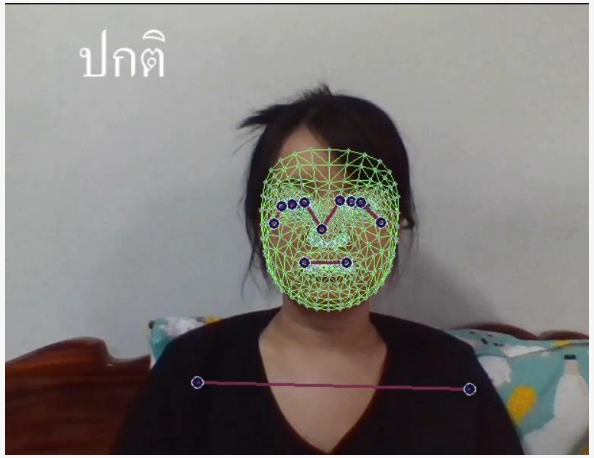
	precision	recall	f1-score	support
หิว	0.67	0.67	0.67	6
ยิ้ม	1.00	1.00	1.00	5
โกรธ	1.00	1.00	1.00	7
ร้องไห้	0.67	1.00	0.80	2
เสียใจ	1.00	1.00	1.00	6
สนุก	1.00	1.00	1.00	5
ขอโทษ	0.80	0.50	0.62	8
ขอบคุณ	0.62	0.83	0.71	6
แฟน	0.43	1.00	0.60	3
เพื่อน	0.89	1.00	0.94	8
คน	1.00	1.00	1.00	3
คนชรา	0.67	0.67	0.67	6
เด็ก	1.00	0.60	0.75	5
ทารก	1.00	1.00	1.00	5
ผู้ชาย	1.00	0.50	0.67	2
ผู้หญิง	1.00	1.00	1.00	4
ผู้ใหญ่	1.00	1.00	1.00	3
วัยรุ่น	0.83	0.83	0.83	6
หนุ่มสาว	0.67	0.50	0.57	4
น้อง	1.00	0.50	0.67	4
ปกติ	0.88	1.00	0.93	7
accuracy			0.84	105
macro avg	0.86	0.84	0.83	105
weighted avg	0.86	0.84	0.84	105

Discussion and real-time recognition

- Based on our experiments, the GRU is an appropriate RNN model for our task. This model is the fastest training with the best results. Therefore,we applied a GRU model to real-time sign language recognition, predicting every last 30 frames of camera video.







Frame by frame

Slicing N frame

FURTHER IMPROVEMENT

01.

Better recognize similar sign language

Improved poses with low precision and recall for better recognize by colle -cting more data with different camera angles, actors, lighting, etc.

02.

More than 20 sign language

Collecting sign language videos with a wider variety of words to build a database to cover sign language used in everyday life.

03.

Serviceable application

Build an application for the hearing impaired to try and use the results to improve it.

04.

Collect more Non-sign language data

Collecting more non-sign language gestures for creating part of model to distinguish whether gestures should be recognized as sign language or not.