

Module Interface Specification for Mechatronics

Team #20, Team Name

Robert Zhu zhul49

Zifan Meng mengz17

Jiahui Chen chenj194

Kelvin Huynh huynhk12

Runze Zhu zhur25

Mirza Nafi Hasan hasanm21

April 6, 2023

1 Revision History

Date	Version	Notes
January 18, 2023	1.0	Everyone - Initial MIS Draft
April 05, 2023	2.0	Everyone - Revised MIS

2 Symbols, Abbreviations and Acronyms

See SRS Documentation at <https://github.com/kelhuynh/OpenASL/blob/main/docs/SRS/SRS.pdf>

Contents

1	Revision History	i
2	Symbols, Abbreviations and Acronyms	ii
3	Introduction	1
4	Notation	1
5	Module Decomposition	1
6	MIS of Motion Tracking Module (M1)	2
6.1	Module	2
6.2	Uses	2
6.3	Syntax	2
6.3.1	Exported Constants	2
6.3.2	Exported Access Programs	2
6.4	Semantics	2
6.4.1	State Variables	2
6.4.2	Environment Variables	2
6.4.3	Assumptions	2
6.4.4	Access Routine Semantics	2
7	MIS of Coordinate Normalization Module (M2)	4
7.1	Module	4
7.2	Uses	4
7.3	Syntax	4
7.3.1	Exported Constants	4
7.4	Semantics	4
7.4.1	State Variables	4
7.4.2	Environment Variables	4
7.4.3	Assumptions	4
7.4.4	Access Routine Semantics	4
7.4.5	Local Functions	4
8	MIS of Coordinate Export Module (M3)	5
8.1	Module	5
8.2	Uses	5
8.3	Syntax	5
8.3.1	Exported Constants	5
8.4	Semantics	5
8.4.1	State Variables	5
8.4.2	Environment Variables	5

8.4.3	Assumptions	5
8.4.4	Local Functions	5
9	MIS of Video Capture Module (M4)	6
9.1	Module	6
9.2	Uses	6
9.3	Syntax	6
9.3.1	Exported Constants	6
9.4	Semantics	6
9.4.1	State Variables	6
9.4.2	Environment Variables	6
9.4.3	Assumptions	6
10	MIS of Video Analysis Module (M5)	7
10.1	Module	7
10.2	Uses	7
10.3	Syntax	7
10.3.1	Exported Access Programs	7
10.4	Semantics	7
10.4.1	State Variables	7
10.4.2	Environment Variables	7
10.4.3	Assumptions	7
10.4.4	Access Routine Semantics	7
10.4.5	Local Functions	8
11	MIS of Keypoint Classification Module (M6)	9
11.1	Module	9
11.2	Uses	9
11.3	Syntax	9
11.3.1	Exported Constants	9
11.4	Semantics	9
11.4.1	State Variables	9
11.4.2	Environment Variables	9
11.4.3	Assumptions	9
11.4.4	Local Functions	9
12	MIS of Machine Learning Module (M7)	10
12.1	Module	10
12.2	Uses	10
12.3	Syntax	10
12.3.1	Exported Constants	10
12.4	Semantics	10
12.4.1	State Variables	10

12.4.2	Environment Variables	10
12.4.3	Assumptions	10
13	MIS of Training Module (M8)	11
13.1	Module	11
13.2	Uses	11
13.3	Syntax	11
13.3.1	Exported Constants	11
13.3.2	Exported Access Programs	11
13.4	Semantics	11
13.4.1	State Variables	11
13.4.2	Environment Variables	12
13.4.3	Assumptions	12
13.4.4	Access Routine Semantics	12
14	MIS of Data Collection Module (M9)	13
14.1	Module	13
14.2	Uses	13
14.3	Syntax	13
14.3.1	Exported Constants	13
14.3.2	Exported Access Programs	13
14.4	Semantics	13
14.4.1	State Variables	13
14.4.2	Environment Variables	13
14.4.3	Assumptions	13
14.4.4	Access Routine Semantics	14
15	MIS of Dynamic Gesture Detection Module (M10)	15
15.1	Module	15
15.2	Uses	15
15.3	Syntax	15
15.3.1	Exported Constants	15
15.3.2	Exported Access Programs	15
15.4	Semantics	15
15.4.1	State Variables	15
15.4.2	Environment Variables	15
15.4.3	Assumptions	16
15.4.4	Access Routine Semantics	16
16	MIS of Dynamic Machine Learning Module (M11)	17
16.1	Module	17
16.2	Uses	17
16.3	Syntax	17

16.3.1	Exported Constants	17
16.3.2	Exported Access Programs	17
16.4	Semantics	17
16.4.1	State Variables	17
16.4.2	Environment Variables	17
16.4.3	Assumptions	18
16.4.4	Access Routine Semantics	18

3 Introduction

The following document details the Module Interface Specifications for OpenASL, a device developed with the aim of translating sign language into text-to-speech, with the purpose of helping members of the deaf and mute community communicate with those who do not know sign language.

Complementary documents include the System Requirement Specifications and Module Guide. The full documentation and implementation can be found at <https://github.com/kelhuynh/OpenASL/>.

4 Notation

N/A

5 Module Decomposition

The following table is taken directly from the Module Guide document for this project.

Level 1	Level 2
Hardware-Hiding Module	
Behaviour-Hiding Module	Text-to-Speech Module Key Point Classification Module - Communicates with ML module with data from coordinate normalization module Training Module - Communicates with ML module to update dataset Coordinate Export Module - Read data from video capture and stores into file Motion Tracking Module - Controller (ties everything together)
Software Decision Module	Video Analysis Module - requires data to be used Machine Learning Module Coordinate Normalization Module

Table 1: Module Hierarchy

6 MIS of Motion Tracking Module (M1)

6.1 Module

motionTrack

6.2 Uses

Video Capture, Coordinate Normalization, Coordinate Export, Video Analysis, Keypoint Classification, TTS

6.3 Syntax

6.3.1 Exported Constants

Name	In	Out	Exceptions
results	image	Object	-
hand_landmarks	-	Tuple of tuples	-
handedness	-	R	-

6.3.2 Exported Access Programs

Name	In	Out	Exceptions
motionTrack	-	-	cv2.error

6.4 Semantics

6.4.1 State Variables

None

6.4.2 Environment Variables

f - file variable for coordinate export purposes

6.4.3 Assumptions

None

6.4.4 Access Routine Semantics

motionTrack():

- output: video consisting of overlay for hand gesture classification into ASL
- exception: `exc := cv2.error`

7 MIS of Coordinate Normalization Module (M2)

7.1 Module

Coordinate Normalization

7.2 Uses

Video Capture

7.3 Syntax

7.3.1 Exported Constants

Name	In	Out	Exceptions
pre_processed_landmark_list	landmark_list	Tuple of tuples	-

7.4 Semantics

7.4.1 State Variables

None

7.4.2 Environment Variables

None

7.4.3 Assumptions

None

7.4.4 Access Routine Semantics

pre_process_landmark(landmark_list):

- output: tuple of 20 tuples consisting of x and y coordinates for each hand joint
- exception: exc := ListIndexOutOfBounds

7.4.5 Local Functions

- __calc_landmark_list

8 MIS of Coordinate Export Module (M3)

8.1 Module

Coordinate Export

8.2 Uses

Coordinate Normalization

8.3 Syntax

8.3.1 Exported Constants

Name	In	Out	Exceptions
keypoint.csv	-	File containing normalized co-ordinates	-

8.4 Semantics

8.4.1 State Variables

None

8.4.2 Environment Variables

None

8.4.3 Assumptions

None

8.4.4 Local Functions

- `__make_csv`

9 MIS of Video Capture Module (M4)

9.1 Module

Video Capture

9.2 Uses

None

9.3 Syntax

9.3.1 Exported Constants

Name	In	Out	Exceptions
success	-	R	-
image	-	Object	-

9.4 Semantics

9.4.1 State Variables

None

9.4.2 Environment Variables

success - indicates that the camera input is initialized for use

9.4.3 Assumptions

There is an available camera connected to the system

10 MIS of Video Analysis Module (M5)

10.1 Module

Video Analysis

10.2 Uses

Coordinate Normalization, Video Capture

10.3 Syntax

10.3.1 Exported Access Programs

Name	In	Out	Exceptions
draw_bounding_rect	-	image	-
draw_landmarks	-	image	-
draw_info_text	-	image	-

10.4 Semantics

10.4.1 State Variables

None

10.4.2 Environment Variables

None

10.4.3 Assumptions

None

10.4.4 Access Routine Semantics

draw_bounding_rect(self, use_brect, image, brect):

- output: image with overlaid bounding rectangle around hand
- exception: exc := None

draw_landmarks(self, image, landmark_point):

- output: image with overlaid hand joints and connections
- exception: exc := None

`draw_info_text(self, image, brect, handedness, hand_sign_text):`

- output: image with overlaid classifier label
- exception: `exc := None`

10.4.5 Local Functions

- `_calc_bounding_rect`

11 MIS of Keypoint Classification Module (M6)

11.1 Module

Keypoint Classification

11.2 Uses

Coordinate Export, Machine Learning

11.3 Syntax

11.3.1 Exported Constants

Name	In	Out	Exceptions
result_index	landmark _{<i>list</i>}	R	ListIndexOutOfRange

11.4 Semantics

11.4.1 State Variables

None

11.4.2 Environment Variables

None

11.4.3 Assumptions

None

11.4.4 Local Functions

- `__call__`

12 MIS of Machine Learning Module (M7)

12.1 Module

ML Train

12.2 Uses

Coordinate Export

12.3 Syntax

12.3.1 Exported Constants

Name	In	Out	Exceptions
keypoint_classifier.hdf5	-	Hierarchical Data Format file	-
keypoint_classifier.tflite	-	ML model	-

12.4 Semantics

12.4.1 State Variables

None

12.4.2 Environment Variables

None

12.4.3 Assumptions

None

13 MIS of Training Module (M8)

13.1 Module

UI Module

13.2 Uses

Video Capture, TTS

13.3 Syntax

13.3.1 Exported Constants

Name	In	Out	Exceptions
images	Video Capture	-	-
frames	Video Capture	-	-
mode	Key Input	-	-
text_string	Machine Model	-	-

13.3.2 Exported Access Programs

-

13.4 Semantics

13.4.1 State Variables

spot: a global variable that keeps track of the position in the text string where the line should be split into two lines. It is initially set to 0.

short: a boolean variable that determines whether the second line of text should be displayed. It is initially set to True.

text: a string that contains the FPS and resolution of the video input. It is constructed based on the image parameter.

line1: a string that contains the first line of the text to be displayed on the UI.

line2: a string that contains the second line of the text to be displayed on the UI.

text_size: a tuple that contains the size of the text to be displayed.

text_w: an integer that contains the width of the text to be displayed.

text_h: an integer that contains the height of the text to be displayed.

multiplier: an integer that determines how many lines of text should be displayed. It is initially set to 1.

13.4.2 Environment Variables

None

13.4.3 Assumptions

None

13.4.4 Access Routine Semantics

None

14 MIS of Data Collection Module (M9)

14.1 Module

DataCollection

14.2 Uses

Coordinate Normalization, Coordinate Export

14.3 Syntax

14.3.1 Exported Constants

Name	In	Out	Exceptions
actions	Number of Gestures	-	-

14.3.2 Exported Access Programs

-

14.4 Semantics

14.4.1 State Variables

No_sequences: Represents the number of sequences to be recorded for each action

14.4.2 Environment Variables

Dirmax: Stores the highest amount of files within a gesture

DATA_PATH: a list of directories where executable files are located

14.4.3 Assumptions

None

14.4.4 Access Routine Semantics

UserInput(): Takes user input and returns a string “action” and checks if action is present in the action list. If it is, it finds the max value of the directories present in the path and creates a directory with a name greater than the max value

15 MIS of Dynamic Gesture Detection Module (M10)

15.1 Module

Gesture Detection

15.2 Uses

Video Capture, Coordinate Normalization, Video Analysis

15.3 Syntax

15.3.1 Exported Constants

-

15.3.2 Exported Access Programs

Name	In	Out	Exceptions
<i>mediapipe_{detection}</i>	Image: frame from CV2 Model: Medi- apipe model that processes the image frame	Image: Pro- cessed Image frame Results: Pro- cessed Data from Mediapipe model	cv2.error

15.4 Semantics

15.4.1 State Variables

Image: refers to a single frame of a video, which is read from a cv2 object

15.4.2 Environment Variables

Cap: an instance of the cv2.VideoCapture class, which is used to capture video from a camera device connected to the system

15.4.3 Assumptions

None

15.4.4 Access Routine Semantics

`mediapipe_detection()`: takes image frame and model as input and processes the image frame through the model, and returns the processed image frame and results to make detections in the video feed

`sel_mode()`: takes a key and mode as input to change between different modes of operation

`cap.read()` used to read frames from the video capture device

16 MIS of Dynamic Machine Learning Module (M11)

16.1 Module

machineLearning

16.2 Uses

Keypoint Classification, Coordinate Export, Coordinate Normalization, Training

16.3 Syntax

16.3.1 Exported Constants

-

16.3.2 Exported Access Programs

-

16.4 Semantics

16.4.1 State Variables

X_train: Numpy array containing the training data for the machine learning model.

X_test: Numpy array containing the testing data for the machine learning model.

y_train: Numpy array containing the labels for the training data.

y_test: Numpy array containing the labels for the testing data.

model: Keras sequential model object representing the machine learning model. Its weights are updated during training.

16.4.2 Environment Variables

None

16.4.3 Assumptions

None

16.4.4 Access Routine Semantics

`np.loadtxt()`: loads gesture labels into an array.

`os.listdir()`: lists the contents of a directory.

`np.load()`: loads a NumPy array from a binary file.

`train_test_split()`: splits the dataset into training and testing sets.

`Sequential()`: initializes the machine learning model.

`model.add()`: adds layers to the machine learning model.

`model.compile()`: configures the machine learning model for training.

`model.fit()`: trains the machine learning model on the dataset.

`model.predict()`: uses the trained model to make predictions on the test set.

`np.argmax()`: gets the indices of the maximum values along an axis of a NumPy array.

`multilabel_confusion_matrix()`: computes a confusion matrix for multiclass classification.

`model.save()`: saves the trained machine learning model to a file.