

Project 77: mmWave HAR

Beck Busch & Sam Mason

Human Activity Recognition

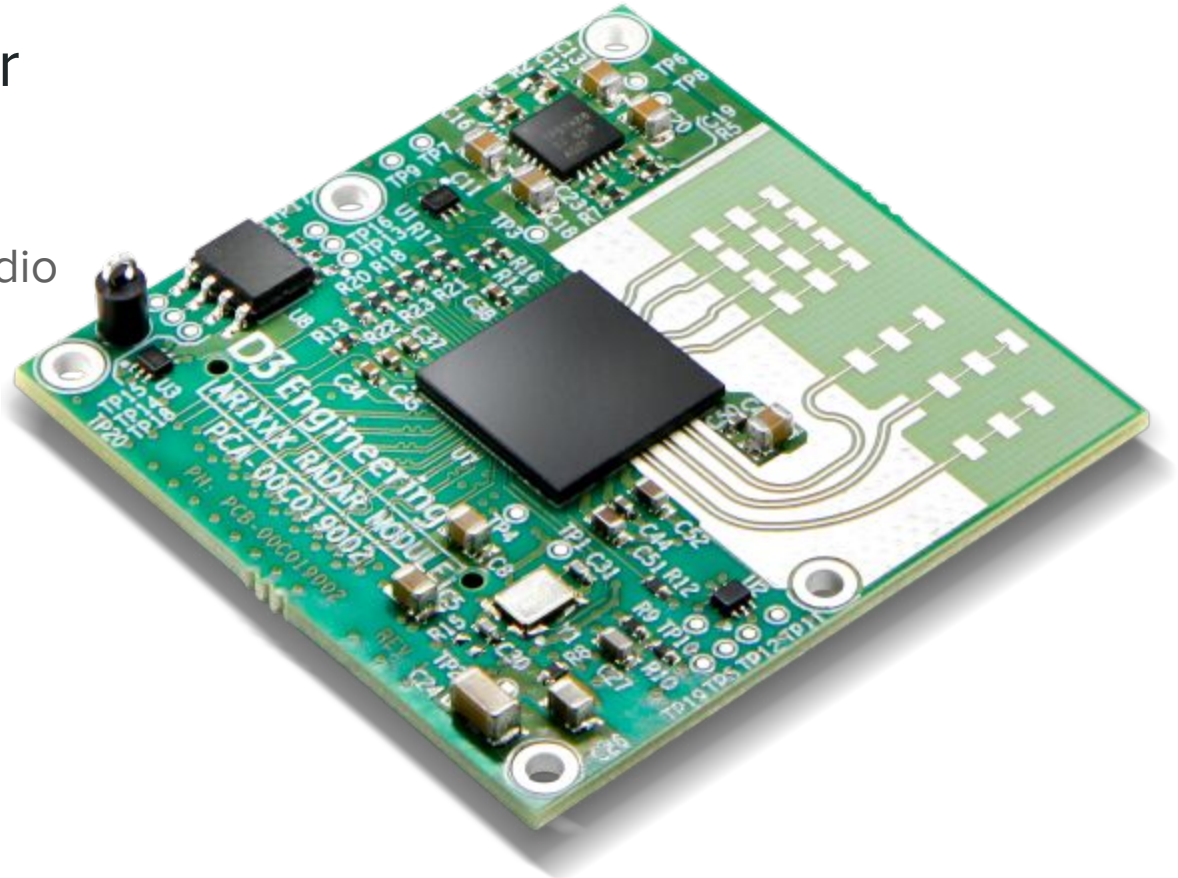
- Identification of human actions and behaviour
 - Camera Tracking
 - Motion Sensors
 - Fitbit / Fitness Watches
- Many Important Uses
 - High Risk Work Environments
 - Smart Home Control
 - Surveillance Systems
 - Medical & Lifestyle Care

Current Approaches

- Optical Sensors
 - Easy to process information
 - Cheap and versatile tech
 - Lighting Conditions
 - Environmental Interference
 - Privacy Concerns
- CSI
 - Readily Available Signal Technology
 - Can Utilize Existing Signals
 - Low Resolution Data
 - Signals Easily Disrupted

Millimeter Wave Radar

- Non Optical Sensor
- Millimeter Wavelength Radio
- Utilizes Doppler Effect



Our Research

- benefits of Human Activity Recognition
 - without the downsides of Optical Sensors
-
- Real-World Context
 - Complex Activity Sequences
-
- Privacy concerns addressed
 - Robust in varying conditions

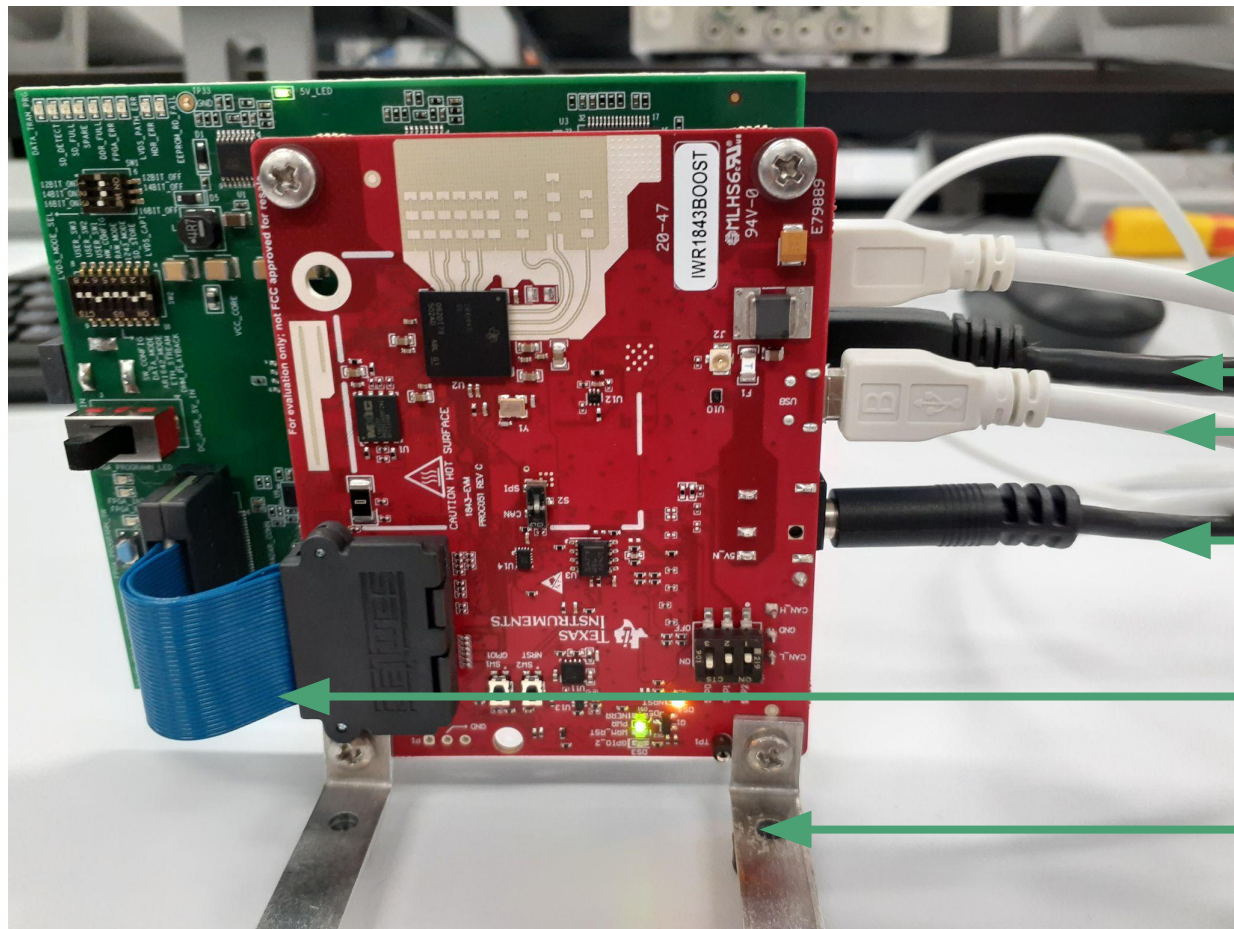
Experimental Survey

- Ethics Review
- Privacy Concerns
- Data Storage
- Participant Comfort

Radar Data Formats

- Point Cloud
- RDI
- RAI
- RDAT
- Combination

Radar Setup



DCA1000EVM
USB

Ethernet Cable

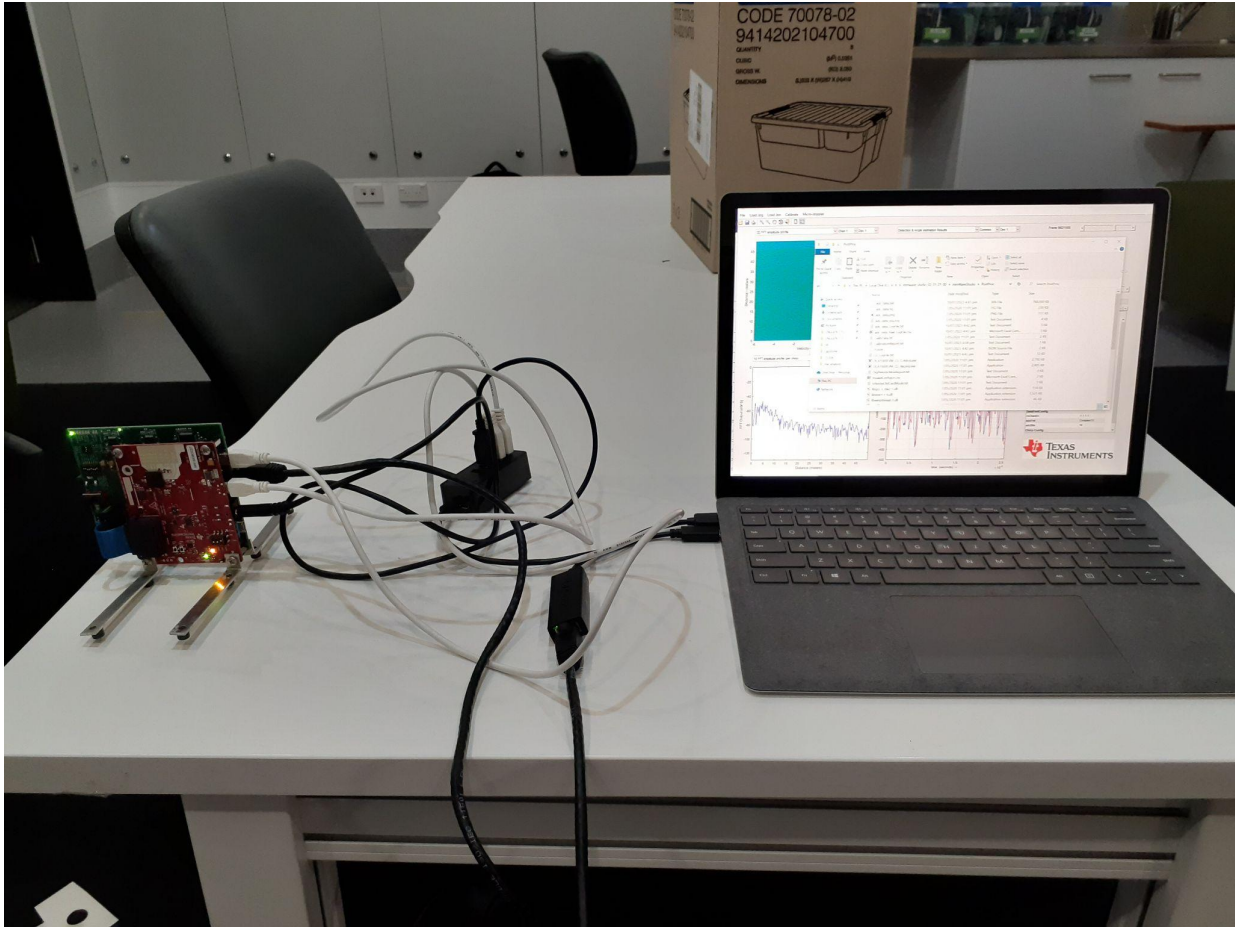
BOOST USB

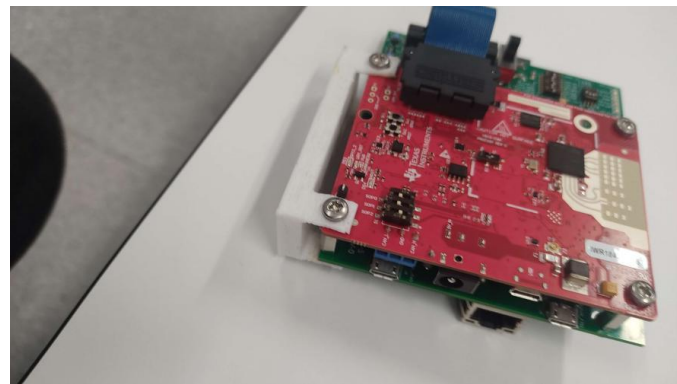
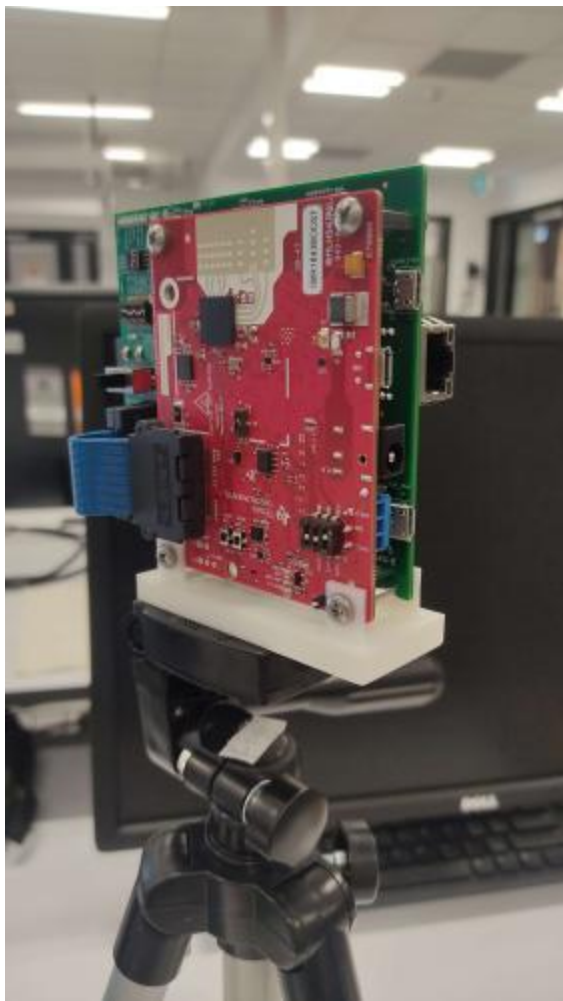
Power

Samtec Ribbon
Cable

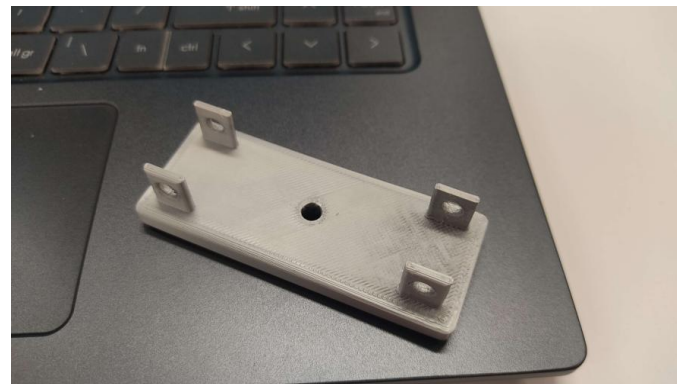
Mounting
Brackets

Radar Setup Cont.



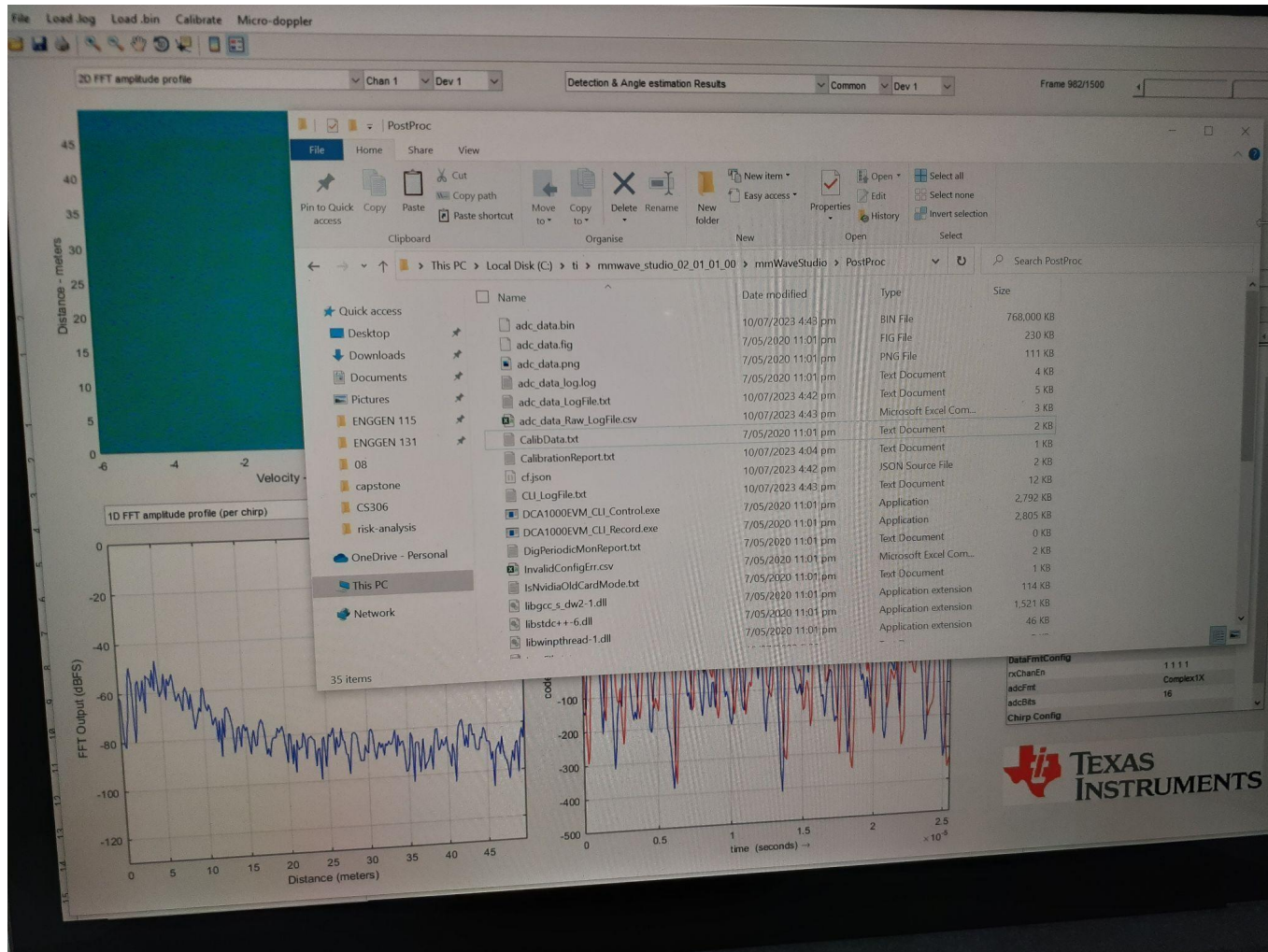


Radar
Setup Cont.

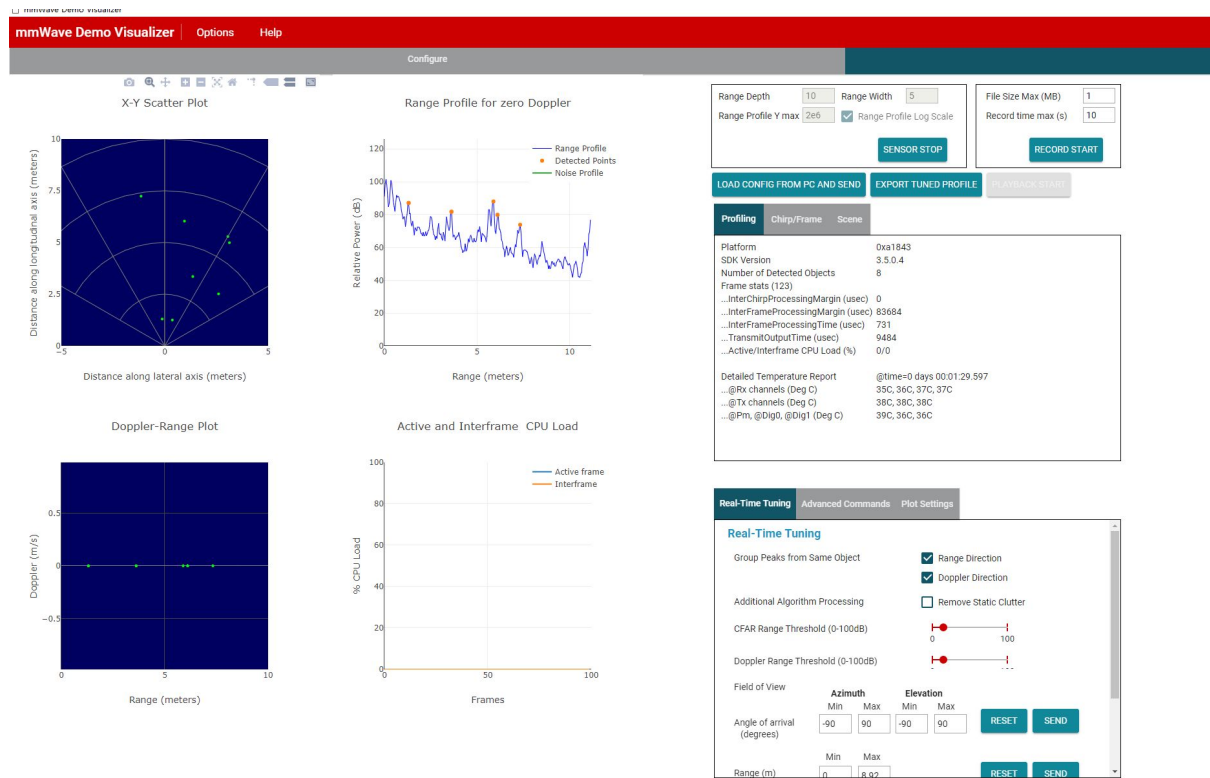


Radar

Save Data



mmWave Demo Visualiser



mmWave Studio

The screenshot displays the mmWave Studio 2.1.1.0 application window. The main interface is divided into several sections:

- Board Control:** Includes a "Reset Control" button and a "Set (1)" button. A note states "SOP Mode controlled via jumper on EV1".
- RS232 Operations:** Features a "COM Port" dropdown set to "COM5" and a "Baud Rate" dropdown set to "921600". A "Disconnected (2)" button is also present.
- Operating Frequency:** Radio buttons for "60 GHz" and "77 GHz" (selected).
- Device Variant:** Radio buttons for "XWR12xx", "XWR6843", "XWR14xx" (selected), and "XWR16xx".
- Files:** Includes fields for "BSS FW:" and "MSS FW:" with file paths, and a "Config File:" field. Buttons for "Load (3)", "Load (4)", and "Load" are provided.
- SPI Operations:** Includes buttons for "SPI Disconnected (5)" and "RF Powered-up (6)".
- Device Information:** Displays details such as "No. of Devices Detected: 1", "FTDI Connectivity Status: Connected", "RS232 Connectivity Status: Connected", "SPI Connectivity Status: Connected", "Device Status: XWR1843/QM/SOP:2/ES:2", "Die Id: Lot4130166/Wafer:3/DevX:18/DevY:34", "BSS firmware version: 2.0.0.1 (05/10/17)", "BSS Patch firmware ver: 1.2.5.2 (30/04/19)", "MSS firmware version: 1.2.5.2 (16/07/19)", "MSS Patch firmware ver: NA", "GUI Version: 2.1.1.0", "Radar Link Version: 2.0.9.0 (31/07/19)", and "Post Proc Version: 4.86".
- Output Window:** Displays a log of messages from the RadarAPI, including commands like "ar1.DownloadBSSFW", "ar1.DownloadMSSFW", "ar1.GetBSSFWVersion", "ar1.GetBSSPatchFWVersion", "ar1.DownloadMSSFW", "ar1.GetMSSFWVersion", "ar1.MSSFWVersion", "ar1.PowerOn(0, 1000, 0, 0)", "ar1.Status: Passed", "MSS power up done async event received!", "ar1.SelectChipVersion('AR1642')", "ar1.SelectChipVersion('XWR1843')", "ar1.SelectChipVersion('XWR1843')", "ar1.Status: Passed", "Device Status : XWR1843/QM/SOP:2/ES:2", "ar1.RFEnable()", "BSS power up done async event received!", "ar1.Status: Passed", "ar1.GetMSSFWVersion()", "ar1.MSSFWVersion: (01.02.05.02 (16/07/19))", "ar1.GetBSSFWVersion()", "ar1.BSSFWVersion: (02.00.00.01 (05/10/17))", "ar1.GetBSSPatchFWVersion()", "ar1.BSSPatchFWVersion: (01.02.05.02 (30/04/19))".

The bottom of the window shows a "Run" button and a "TEXAS INSTRUMENTS" logo.

mmWave Studio Cont.

The screenshot displays the mmWave Studio 2.1.1.0 software interface. The main window is titled "mmWave Studio 2.1.1.0" and features a menu bar with "File", "View", "Tools", "ToolBars", "Window", and "Help". Below the menu bar is a toolbar with icons for "RadarAPI", "MSSMon", "DynamicChirpCfg", "ClockOutCfg", "CalibDataCfg", and "Import_Export".

The interface is divided into several configuration panels on the left and a log output window on the right.

Static Configuration

- Basic Configuration**
 - Channel Config**
 - Tx Channel**: ☒ Tx0 ☒ Tx1 ☒ Tx2
 - Rx Channel**: ☒ Rx0 ☒ Rx1 ☒ Rx2 ☒ Rx3
 - Cascading Mode**: Single Chip
 - CasCading PinOut Cfg**
 - ☐ ClkOut Master Dis ☐ SyncOut Master Dis
 - ☐ ClkOut Slave Ena ☐ SyncOut Slave Ena
 - ☐ INTLO Master Ena ☐ OSCClkOut Master Dis
 - ADC Config**
 - Bits**: 16
 - Full Scale Reduction Factor**: 0
 - Format**: ComplexTx
 - IQ Swap**: IFirst
- Advanced Configuration**
 - RF LDO Bypass**
 - ☐ RF LDO Bypass Enable
 - ☐ PALDO IP Disable
 - Supply IR Drop**: 0%
 - IO Supply**: 3.3
 - LP Mode**
 - LP ADC Mode**: RegularADC
 - Radar Miscellaneous Control**
 - ☐ Per Chirp Phase Shifter En
- Frequency Limits Configuration**
 - Frequency Limit Low (GHz)**: 77.0
 - Frequency Limit High (GHz)**: 81.0
- Cal Mon Frequency TX Power Limits Config**

	Tx0	Tx1	Tx2
Freq Limit Low	77.00	77.00	77.00
Freq Limit High	81.00	81.00	81.00
Power Backoff	0	0	0

The right-hand side of the interface shows the "Output" window, which displays a log of script messages. The log includes various status messages and data updates, such as:

- [15:07:32] [RadarAPI]: ari.PowerOn(0, 1000, 0, 0)
- [15:07:32] [RadarAPI]: Status: Passed
- [15:07:32] MSS power up done async event received!
- [15:07:36] [RadarAPI]: ari.SelectChipVersion("PAR1642")
- [15:07:36] [RadarAPI]: Status: Passed
- [15:07:36] [RadarAPI]: ari.SelectChipVersion("XWR1843")
- [15:07:36] [RadarAPI]: Status: Passed
- [15:07:36] Device Status : XWR1843/QM/SOP:2/ES:2
- [15:07:36] [RadarAPI]: ari.RfEnable()
- [15:07:36] BSS power up done async event received!
- [15:07:36] [RadarAPI]: Status: Passed
- [15:07:36] [RadarAPI]: ari.GetMSSFWVersion()
- [15:07:37] [RadarAPI]: MSSFWVersion: (01.02.05.02 (16/07/19))
- [15:07:37] [RadarAPI]: ari.GetBSSFWVersion()
- [15:07:37] [RadarAPI]: BSSFWVersion: (02.00.00.01 (05/10/17))
- [15:07:38] [RadarAPI]: ari.GetBSSPatchFWVersion()
- [15:07:38] [RadarAPI]: BSSPatchFWVersion: (01.02.05.02 (30/04/19))
- [15:12:35] [RadarAPI]: ari.ChanNAdoConfig(1, 1, 1, 1, 1, 1, 1, 2, 1, 0)
- [15:12:35] [RadarAPI]: Status: Passed
- [15:12:43] [RadarAPI]: ari.ChanNAdoConfig(1, 1, 1, 1, 1, 1, 1, 2, 1, 0)
- [15:12:43] [RadarAPI]: Status: Passed
- [15:13:17] [RadarAPI]: ari.LPModConfig(0, 0)
- [15:13:17] [RadarAPI]: Status: Passed
- [15:13:21] [RadarAPI]: ari.RfInit()
- [15:13:22] RF Init async event received!
- [15:13:22] [RadarAPI]: Status: Passed
- [15:13:22] [RadarAPI]: Time stamp, Temperature: 345483.37; APLL Status, Update: 1, 0; SynthVCO1 Status, Update: 1, 1; SynthVCO2 Status, Update: 1, 1; LODist Status, Update: 1, 1; RxADCC Status, Update: 1, 1; HPFcuttoff Status, Update: 1, 1; LPFcuttoff Status, Update: 1, 1; PeakDetector Status, Update: 1, 1; TxPower Status, Update: 1, 1; RxGain Status, Update: 1, 1; TxPhase Status, Update: 1, 1; RxIQMM Status, Update: 1, 1;

The bottom of the interface features a "Run" button and a "Pause" button, along with a "Browse" button and the Texas Instruments logo.

The screenshot displays the mmWave Studio 2.11.0 software interface, which is used for configuring and testing radar systems.

RadarAPI Configuration

- Connection:** MSSMon, DynamicChirpCfg, ClockOutCfg, CalibDataCfg, Import_Export
- Sensor Configuration Profile:**
 - Profile Id: 0
 - Start Freq (GHz): 77.000000
 - Frequency Slope (MHz/us): 29.982
 - Idle Time (us): 100.00
 - TX Start Time (us): 0.00
 - ADC Start Time (us): 6.00
 - ADC Samples: 256
 - Sample Rate (kpsps): 10000
 - Ramp End Time (us): 60.00
 - RX Gain (dB): 30
 - RF Gain Target: 30dB
 - VCO Select: VCO1
 - Calib LUT Update: ☐ RetainTxCalLUT ☐ RetainRxCaliLUT
- Chipr:**
 - Profile Id: 0
 - Frequency Slope Var (MHz/us): 0.000
 - Start Chirp for Cfg: 0
 - End Chirp for Cfg: 0
 - Start Freq Var (MHz): 0.000000
 - TX Enable for current chirp: ☒ TX0 ☐ TX1 ☐ TX2
- Frame:**
 - Start Chirp TX: 0
 - End Chirp TX: 0
 - No of Frames: 8
 - Dummy Chips(End): 0
 - Trigger Select: SoftwareTrigger
 - Test Source Enable: ☐

Waveform Capture and Processing

A diagram illustrates the timing sequence for capturing a chirp through the chip configuration RAM. Key events include: Idle Time, ADC Valid Start Time, ADC Sampling Time, Frequency Slope, Ramp End Time, and Trigger Delay. Below the diagram are controls for capturing and processing waveforms:

- Capture and Post Processing:** DCA1000 ARM, Trigger Frame, Stop Frame, Transfer Files, PostProc
- File Path:** C:\mmwave_studio_02_01_01\00\mmWaveStudio\PostProcladc_data_v... Browse

Inter Rx Gain Phase Freq Control Config

	Rx0	Rx1
Dig Gain (dB)	0.0	0.0
Dig Ph Shift (Deg)	0.00	0.00
ProfileIndex	0	

Script Messages Only

```
[16:04:53] [RadarAPI]: Status: Passed
[16:04:57] [RadarAPI]: arl.LvdsClkConfig(1, 1)
[16:04:58] [RadarAPI]: Status: Passed
[16:05:03] [RadarAPI]: arl.ProfileConfig(0, 77, 100, 6, 60, 0, 0, 0, 0, 0, 29.982, 0, 256, 10000, 0, 30)
[16:05:03] [RadarAPI]: Status: Passed
[16:05:04] [RadarAPI]: arl.ChirpConfig(0, 0, 0, 0, 0, 0, 0, 1, 0, 0)
[16:05:04] [RadarAPI]: Status: Passed
[16:05:05] Test Source Already Disabled.....!
[16:05:05] [RadarAPI]: arl.DisableTestSource(0)
[16:05:05] [RadarAPI]: Status: Passed
[16:05:05] [RadarAPI]: arl.FrameConfig(0, 0, 8, 128, 40, 0, 0, 1)
[16:05:05] [RadarAPI]: Status: Passed
```

At the bottom left, there are buttons for "Run!" and "Pause". At the bottom right, there are icons for file operations and the Texas Instruments logo.

Experiment Methodology

- Participant prepares in front of radar
- Given start command as recording begins
- Recording file is processed by mmWave Studio PostProc
- Processed file is moved to external drive and renamed
- Participant given next activity

File Naming Scheme

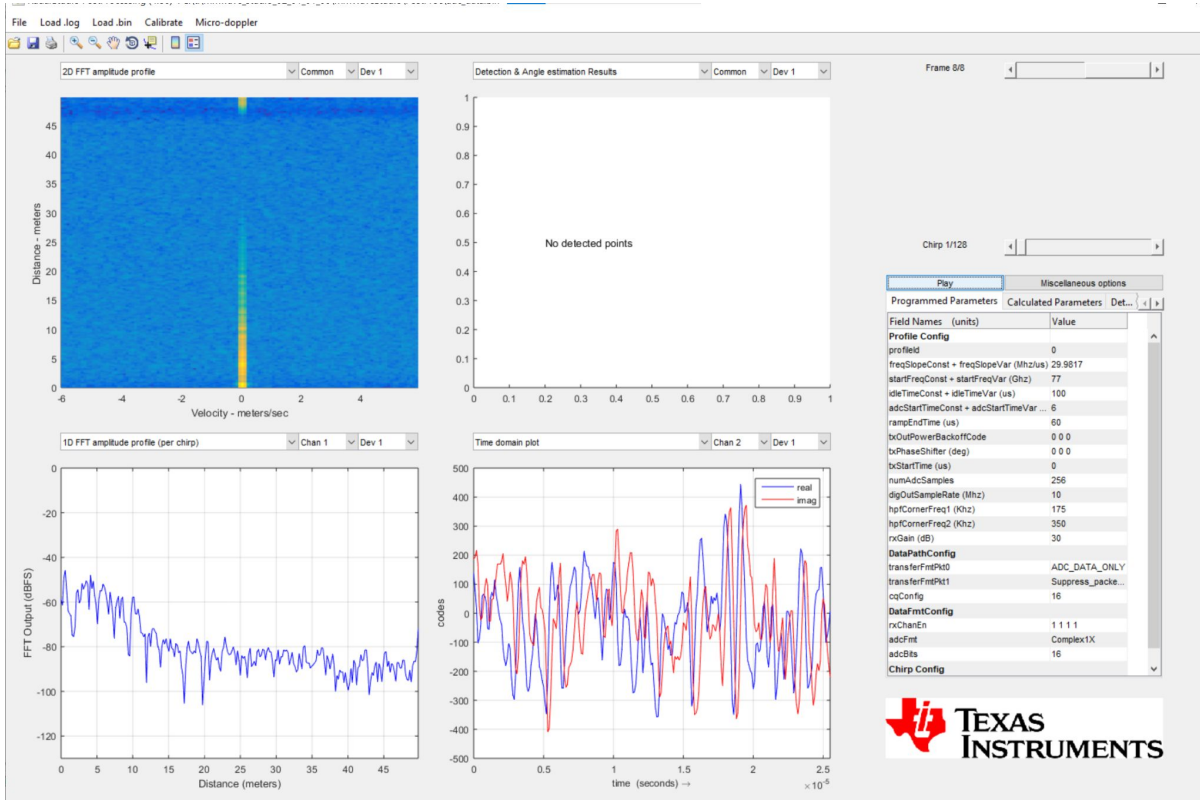
Files should be named according to:

- The activity being performed
- The participant number
- The activity duration
- The activity number according to the activity only (starts at 1)
- The starting timestamp (unix)

Some example file names could look like this:

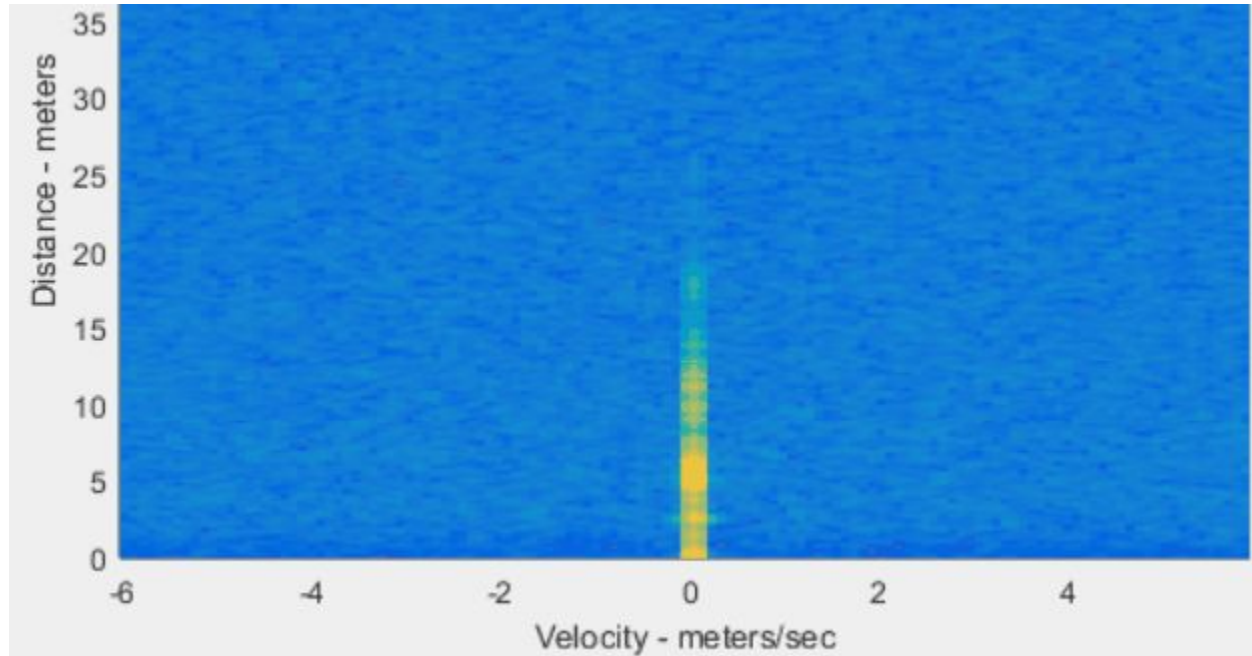
- boxing_01_6s_1_1689832549
- jumping_01_1m4s_3_1689839667
- sitting_standing_walking_03_15s_2_1689833322
- throwing_catching_11_6s_1_1689832676

Data Visualisation

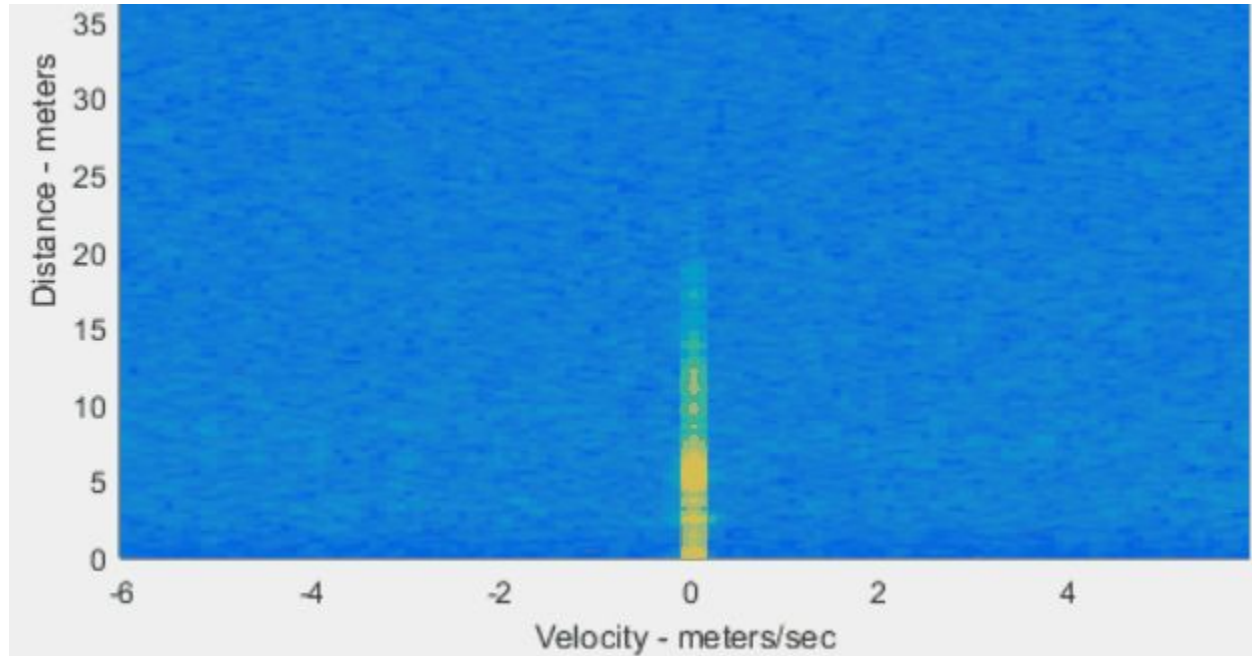


- 2D FFT amplitude profile
- Range-Angle plot (per Frame)
- Detection & Angle estimation Results
- Chirp Config Picture
- 1D FFT amplitude profile (per chirp)
- Time domain plot
- and more...

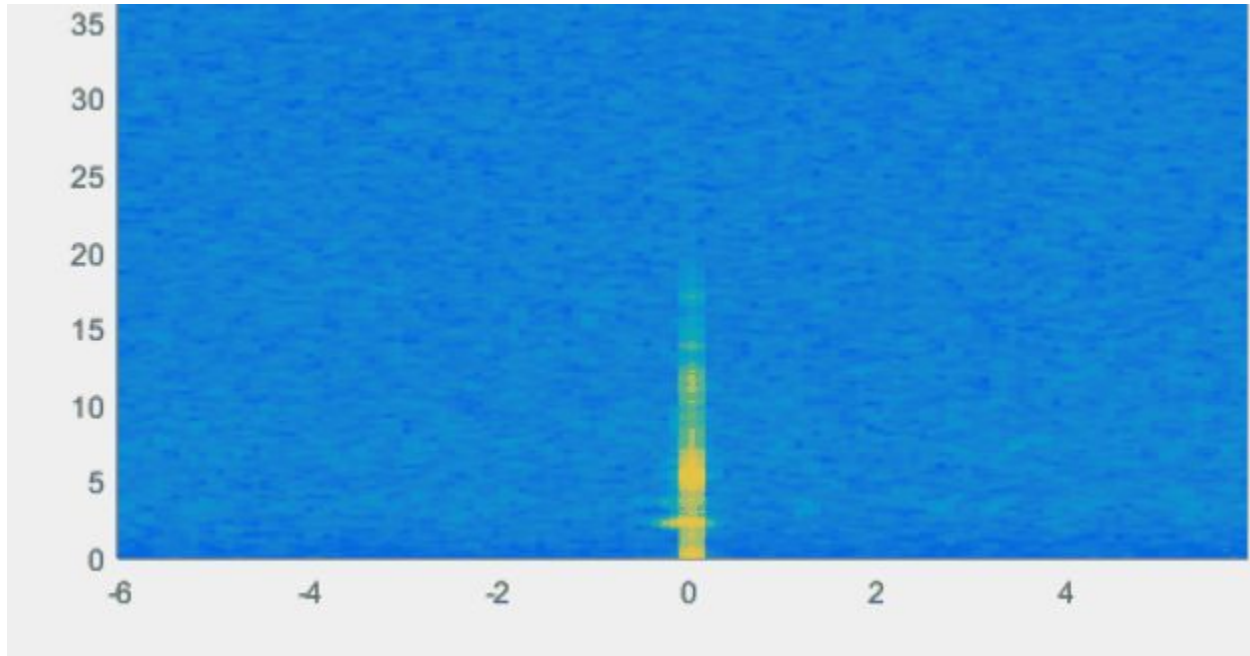
Sample Data Recordings - Walking



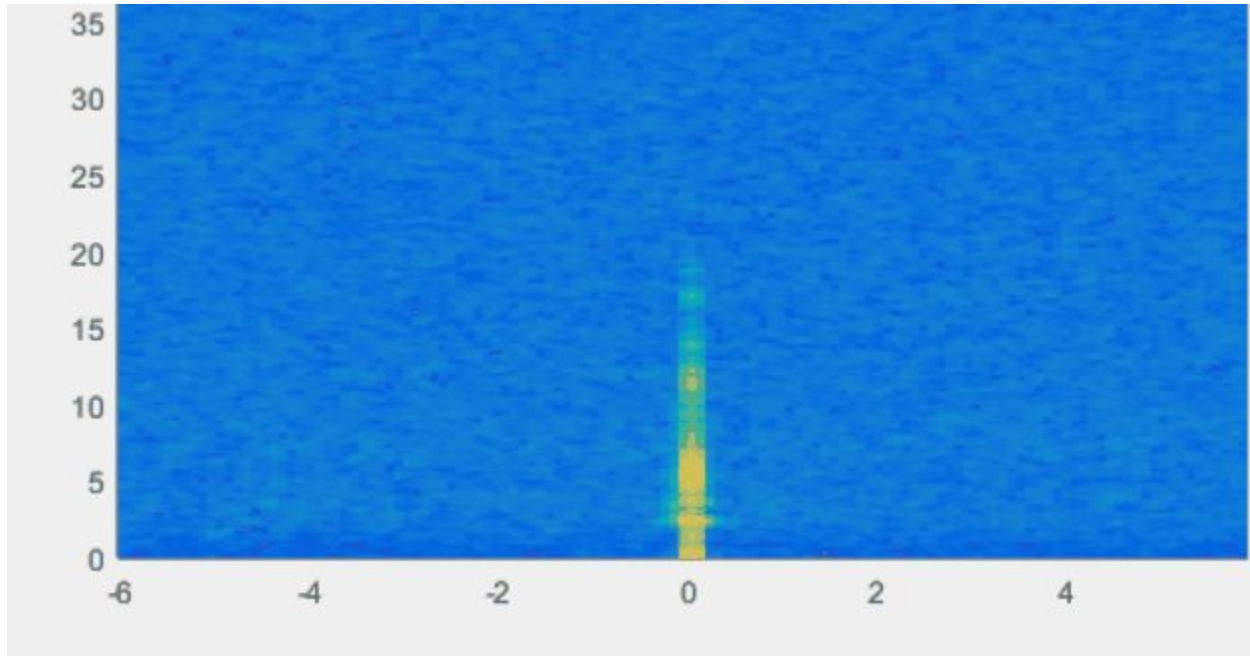
Sample Data Recordings - Reading



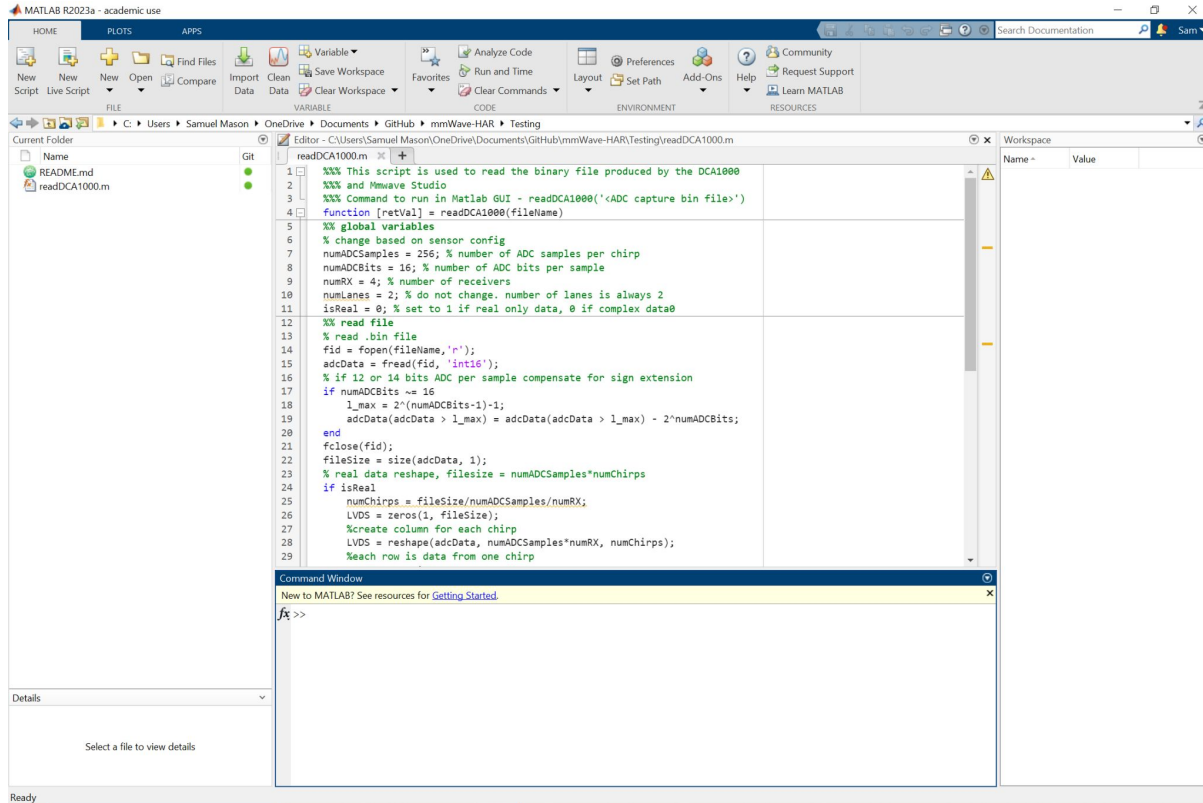
Sample Data Recordings - Jumping



Sample Data Recordings - Boxing



MATLAB Processing



Activity Classes (WIP)

Simple (12)
Sitting
Standing
Walking
Clapping (while standing)
Squatting down
Jumping
Throwing (L + R)
Catching
Reading
Writing
Snapping fingers (L + R)
Pointing (L + R)

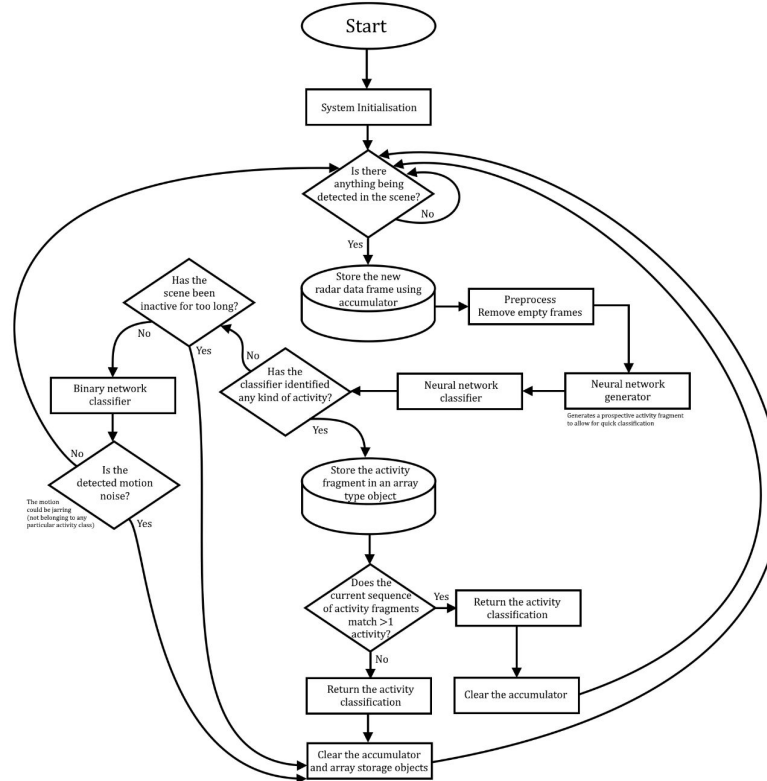
Complex (6)
Stand up then walk (3) [sitting -> standing -> walking]
Standing ovation (2) [sitting -> standing + clapping]
Squat jumps (2) [squatting down -> jumping]
Playing catch (2+) [throwing -> catching OR throwing -> standing -> catching -> standing]
Annotating (2) [reading -> writing OR writing -> reading]
Commanding attention (2) [snapping fingers -> pointing]

Aiming for activities:

- Office
- Living
- Exercise

Processing Flow (WIP)

The flowchart depicts a generative adversarial approach to allow for real-time predictive capabilities.



Machine Learning Model

- HMM
- CNN + LSTM
- TCN
- CGAN
- Other networks?

Real-Time Inferencing

- Explored in a limited capacity by existing literature
- Want to replace static/dynamic segmentation
- Generative capabilities needed?
- Yet to decide if we are to implement