Signal Intelligence module for GNU Radio (gr-sigint)

Introduction

Abstract

The project aims at the development of an out-of-tree module for signal intelligence (gr-sigint) which is able to monitor, automatically classify and consequently demodulate wireless signals. The classification of signals based on their modulation can be achieved by extracting the cyclostationary features from the signals and using standard wireless frequency band allocation information to aid the classification. These features can also be exploited by deep neural networks to help in the classification of modulated signal in a less explicit manner. Another alternative approach to classify the signals can be, by matching the spectral waterfall image of the signal from a database (1) of known waterfall images using image processing/deep learning techniques like CNNs.

Modulation Recognition

- Modulated signals in general have several distinct features like frequency, phase, amplitutde.
- Linear and nonlinear transforms and filtered outputs of these features are specific to modulation types.
- Challenge lies in extracting these features in realtime and analysing them to classify them according to modulation (2).
- Classification methods may suffer from the presence of receiver noise and channel fading which demands for a robust implementation of the recognition module.

The techniques that gr-sigint will use to recognize the modulation are :

- Cyclostationary Features :
  - Modulated signals are cyclostaionary processes.
  - Cyclostationary processes have a periodic autocorelation function.
  - These periodic signals autocorelation and SCFs can be extracted and analyzed using Fourier analysis for moodulation type presence (5).
  - These feature detectors and analyzers though computationally expensive, can be easily implemented in GNU-Radio using existing blocks (filters, FFTs).
- Deep Neural Networks :
  - Deep Neural Networks (DNNs) with its hidden layers helps in learning the complex non-linear relationship between the input and the output.
  - The modulated signals have several complex non-linear properties that are distinct and specific to the modulation used.
  - Due to these properties of DNNs, they can be used to recognize the modulation from signals.
  - They also have improved performance at low SNRs (1).
- Convolutional Neural Networks :
  - The waterfall images of the signals have features and patterns that are fairly distinct and specific to the modulation (4) used.
  - Convulational Neural Networks have proven to be state-of-the-art methods in image classification (ISLRVC, MS COCO) and can be deployed here for modulation recognition as well.
  - CNNs work by finding the optimum layers of self-learning filters using backpropogation on training data, which when applied to the test data give accurate classification results. Thus CNNs can be used to classify the waterfall images of signals.
  - Widely available tools to implement CNNs make it easy to deploy them for modulation recognition.

Project Structure

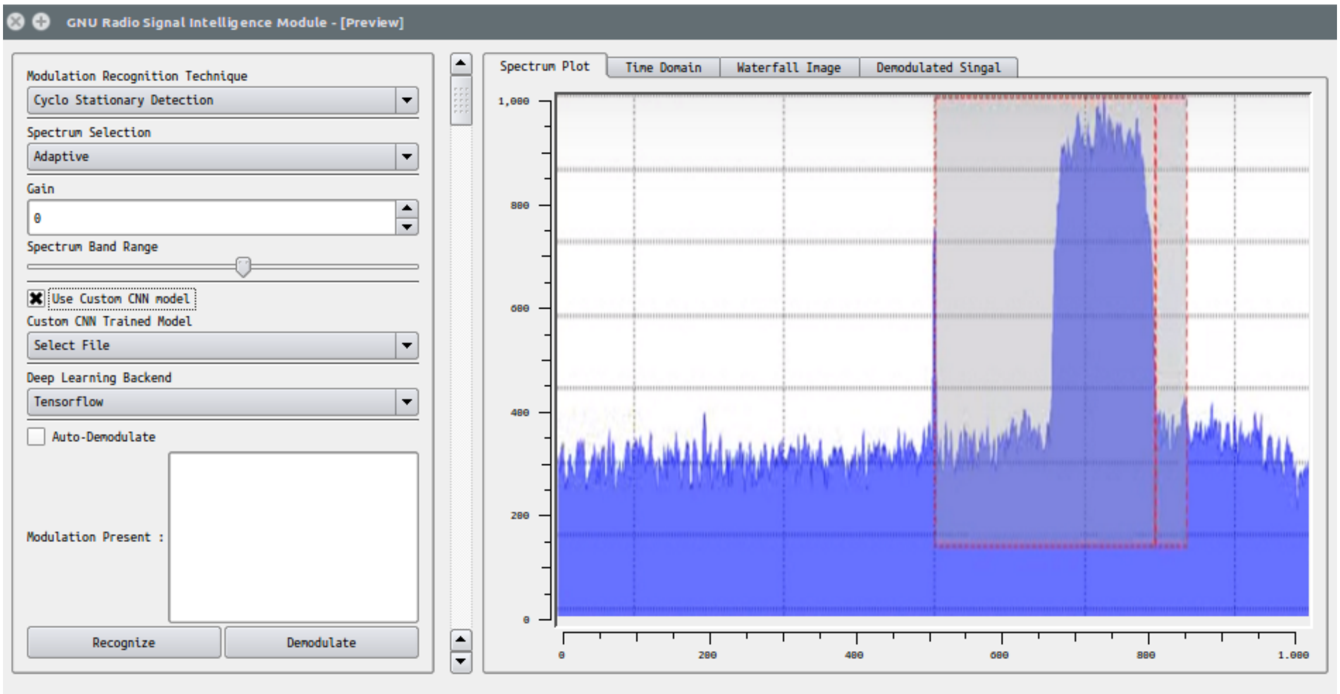
Features and Deliverable Specifications

The module features would be as follows on a very high level:

- Automated Modulation recognition from incoming signals using various feature analysis and deep learning techniques like:

- Cyclostationary feature detection and analysis using existing blocks in gnu-radio.
    - Convolutional Neural Networks (CNNs) using the waterfall images of the incoming signals as input implemented in a deep learning framework(DLF) like Tensorflow/Theano.
    - Convolutional Neural Networks using the modulated signal as used in (1) with input implemented in a deep learning framework(DLF) like Tensorflow/Theano.
    - A few other modulation techniques using DNNs maybe explored in the end as time permits.
- A Python API to connect gr-sigint with required components of deep learning frameworks like Theano/Tensorflow.
- A Qt GUI with options to select the frequency band, modulation recognition technique, radio parameters.
- Coverage :
    - Digital: BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, and PAM4
    - Analog: WB-FM, AM-SSB, and AM-DSB
- Development of a demo for automatic classification that covers simple modulation schemes (FSK,PSKs).
- Tutorials and Documentation for the module.

A primitive GUI for the gr-sigint module:



Project Timeline

This is merely a modest sketch. I have tried to be as lenient as possible in assigning the weekly tasks. The last 1-2 days of a Week will usually be reserved for code review and documentation or buffers to complete work which has not been completed in the allotted time. Though I intend to keep in touch with the mentor throughout the week and ensure that I am going in the right direction, I have tried to maintain a feasible feedback mechanism in the schedule so as to ensure that my mentor gives feedback on a reviewable quantity of code at appropriate intervals.

- **Community Bonding period**:
    - Read and brush up the relevant theory needed to implement the module.
    - Get working knowledge of relevant modules/blocks of gnuradio which will be reused in gr-signint.
- **Week 1 (6th May - 13th May)**:
    - Implementation of GUI.
    - Implementation of a python API which connects GUI with external modulation recognition scripts.
- **Week 2 (13th May - 20th May)**:
    - Begin with the cyclostationary feature approach.
    - Identify existing blocks in gnu-radio to implement cyclostationary feature detectors.
- **Week 3 (20th May - 27th May)**:
    - Implement the feature detectors using existing blocks from gnu-radio.
    - Implement wrappers for modulation recognition around the feature detectors.
- **Week 4 (27th May - 3rd June)**:
    - QA testing for the cyclostationary approach with modulations in coverage set with and without noise.
- **Week 5 (3rd June - 10th June)**:
    - Begin with the CNN-waterfall approach.

- Reuse the waterfall block in gnu-radio to generate waterfall images from incoming signals.
- Generate waterfall image dataset for modulations in coverage set to feed to the CNN.

- **Week 6 (10th June - 17th June)**:
  - Implement the CNN to classify the waterfall images into modulations from coverage set.
  - Implement a python API to classify waterfall images of incoming signals using the trained model from the CNN.
- **Week 7 (17th June - 24th June)**:
  - QA testing for the CNN-waterfall approach with modulations in coverage set with synthetic and real-world data(noisy).
- **Week 8 (24th June - 1st July)**:
  - Begin with the CNN approach as mentioned in (1).
  - Generate signal dataset for modulations in coverage set. Data can be taken from (1) (whichever is earlier).
  - Exhaustive chat with mentors regarding mid-term evaluation and some parts of code that I might need to improve upon.
- **Week 9 (1st July - 8th July)**:
  - Implement the CNNs to classify the signals into modulations from coverage set.
  - Implement a python API to classify incoming signals in gnu-radio using the trained model from the CNN.
- **Week 10 (8th July - 15th July)**:
  - QA testing for the CNN approach with modulations in coverage set with synthetic and real-world data (RTL-SDR/USRP).
- **Week 11 (15th July - 22th July)**:
  - Documentation and Buffer to complete any remaining parts.
- **Week 12 (22th July - 29th July)**:
  - Write tutorials explaining how to use the gr-sigint module over some standard datasets and realtime signals (RTL-SDR/USRP).
  - Ideally a week to complete any remaining documentation of the code.
  - Buffer week for analyzing the suggestions by the gnu-radio community and talking to the mentor regarding implementing the same.
  - Implementing the suggestions.
  - Final evaluation exhaustive chat with mentors and a feedback for the change in the code needed.

Note: The documentation for each week's work shall be done by the end of that week only. In any case, any code shall not stay undocumented for more than 1-2 weeks.

Student Information

Personal Information

- **Name** : Meet Pragnesh Shah
- **City** : Mumbai, India
- **Skype** : meetshah1995
- **Email** : meetshah1995@gmail.com
- **Alternate Email** : meetshah1995@ee.iitb.ac.in
- **Github** : https://github.com/meetshah1995/
- **Website** : http://meetshah1995.github.io
- **Telephone** : +91 916 758 5853
- **Time zone** : GMT +5:30
- **IRC nick** : meetshah1995
- **Primary Language** : English

Education

- **University**: Indian Institute of Technology Bombay (IIT-Bombay)
- **Degree**: B.Tech & M.Tech (Dual Degree) in Electrical Engineering
- **Minors**: Computer Science and Operations Research
- **Master's Specialization**: Communications and Signal Processing

Secret Code 2016 : Polar Codes are the coolest codes

Summer Availability

- Classes End Date: 15/04/2016 (dd-mm-yyyy)
- Exams End Date: 05/05/2016
- Work Hours per week: Approximately 42 hours per week or the amount of time required to finished the tasks allocated for the week (whichever is higher).

Background

Being a communications engineering student and interested in statistical signal analysis, machine learning and image processing, I think this is a very good opportunity for me to get hands on development experience in this field. I am definitely prepared to learn more about it in the summers and explore this interesting field. I think given my interests, I would like to add more functionalities to gnu-radio post GSoC as and when time permits me.

References

1 : Convolutional Radio Modulation Recognition Networks, *T. O'Shea , J Corgan*
2 : Automatic Modulation Classification in Practice, *S. Rajendran*
3 : Software Radio-Based Decentralized Dynamic Spectrum Access Networks, *F. Ge*
4 : Signal Identification Wiki for waterfall images
5 : Cyclostationary Feature Detection, *A. Sahai*
6 : Spectral estimation routines for GNU Radio, *KIT-CEL*