**Magnetic Resonance Imaging Automated Quality Assurance Testing - Project Hazen**

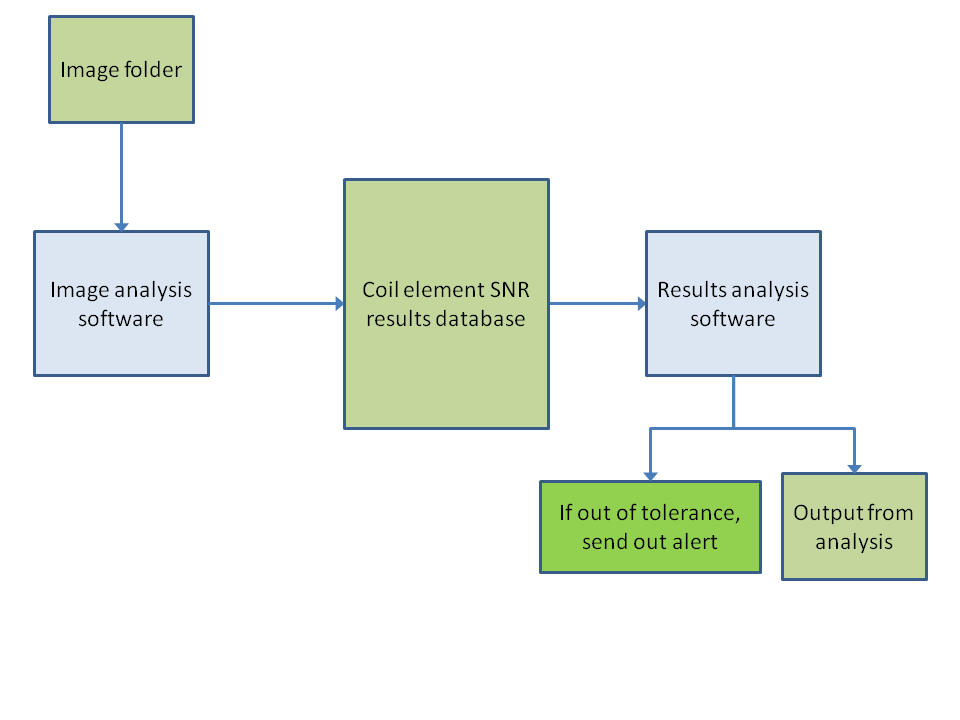
**1. Introduction**

Radio-frequency (RF) coils are a key part of a magnetic resonance imaging (MRI) system. Routine quality assurance (QA) testing of the RF coils is conducted in order to identify faults. RF receiver coils are typically made up of several elements, with the signal from each element being combined to produce a complete image. Minor degradation in the performance of a single coil element may not be obvious in clinical images; coil-QA testing allows each element to be tested individually in a controlled manner, usually following a manufacturer-defined protocol.

Manufacturer coil-QA tests provide assurance that the current performance of a particular coil is within acceptable tolerances, but the data from these tests is not aggregated to identify performance trends over time, or to try and monitor system-wide performance. The purpose of this work was to reanalyse the images being produced by routine coil-QA tests and use them to gather more information about system performance. The end-goal of the work was to produce a software tool that could be set to run on a DICOM server and automatically evaluate coil-QA images as they are produced. The software would then collate longitudinal performance data for these systems and send out alerts if significant drops in performance were identified.

This document describes the structure of the software that has been developed, before presenting the output of the software produced from a test dataset. The suitability of the software and test data is discussed, and recommendations are made for further work.

**2. Project Structure**



*Figure 1: Top-level system diagram. The blue boxes represent software tools that need to be created.*

Figure 1 illustrates how the software for the project fits together. The primary components are as follows:

1. *Image folder*

This is a folder containing all of the DICOM images output by the manufacturer coil-QA tests. For the purposes of this work, this is simply a folder of test images, but it is intended that this folder could be populated with the appropriate QA images as they are received at a DICOM server.

1. *Image analysis software (mr\_coil\_qa.py)*

This is a Python script that reads in the images within the image folder, reads in relevant information from the DICOM header, calculates a value for the image signal to noise ratio (SNR) and saves all this information to an output file.

1. *Coil element SNR results database (coil\_snr\_results.csv)*

This is where the results of the image analysis software are stored. Currently, this is simply implemented as a comma-separated value (csv) file. It is thought that it may be beneficial for a fully-fledged database format (such as SQL) to be used instead, to facilitate further analysis.

1. *Results analysis software (mr\_coil\_analysis.py)*

This is another Python script that reads in the results database (currently a csv file) and analyses the data within it. In its current form, this script collates results that have the same coil, coil element and test slice location, before plotting the SNR results for each group against the test date.

1. *Output from analysis*

This is a folder where the plots generated by the results analysis software are saved.

**3. Image Analysis Code**

The following steps are performed by the main image analysis code:

1. Read the file names from the image folder.
2. Create a set of empty 1D arrays (of the same length as the number of images to be analysed), one for each item of metadata which will be recorded + one for the calculated SNR result. The metadata being collected includes information to uniquely identify the unit being tested and the time/date of the test, as well as information about the image sequence that could influence the SNR (such as the slice thickness, flip angle, TE, TR etc.).

Then, for each image in turn:

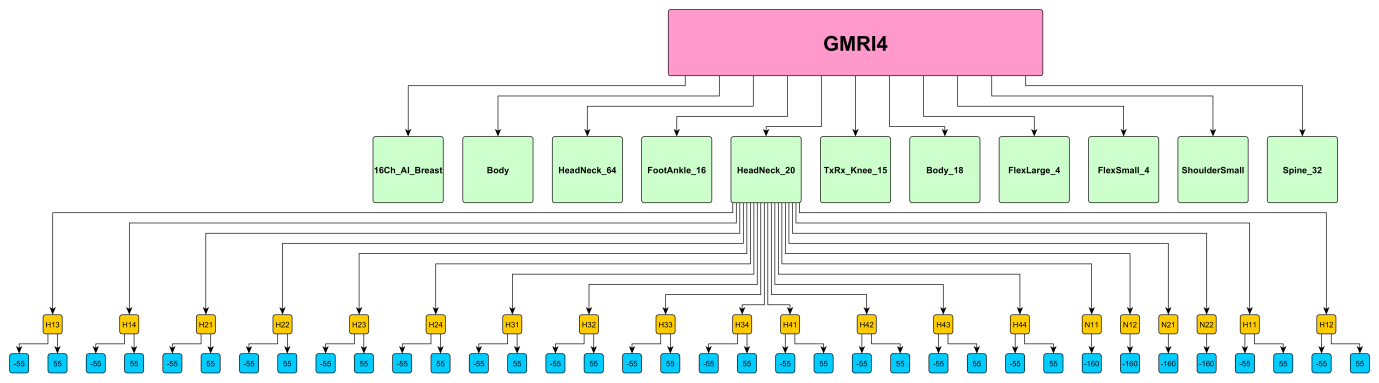
|  |  |
| --- | --- |
| 1. The image files are read and the required metadata is extracted from the file header. The image data is converted into a numpy array for further processing. |  |
| 1. A 9x9 boxcar kernel is used to create a smoothed image. A noise image is also generated by subtracting the smoothed image from the unprocessed image (this will be used when calculating SNR later) . The brightest pixel location in the smoothed image is noted - this should be located at the position on the phantom edge closest to the coil element being tested. |  |
| 1. A binary image is produced by applying a 5x5 pixel median filter on the unprocessed image and then thresholding this image using the triangle method . This binary image is then used to determine the centre of the phantom (by calculating the centre of mass). |  |
| 1. A 20x20 pixel region of interest (ROI) is defined at the midpoint between the brightest location and the phantom centre. The coil-element SNR is calculated as the mean pixel value within this ROI in the smoothed image divided by the standard deviation of the pixel values within this ROI in the noise image. | **SNR = 38.35** |

1. Once all of the images have been analysed, the SNR data, along with the metadata is output into a csv file (if another output file already exists, the new data is appended to this file).

**Data Analysis Case Study - Siemens Skyra 3T - GMRI4**

To test the analysis software, a dataset of coil-QA test images for a single Siemens MRI scanner was used, covering a period of around 10 months.

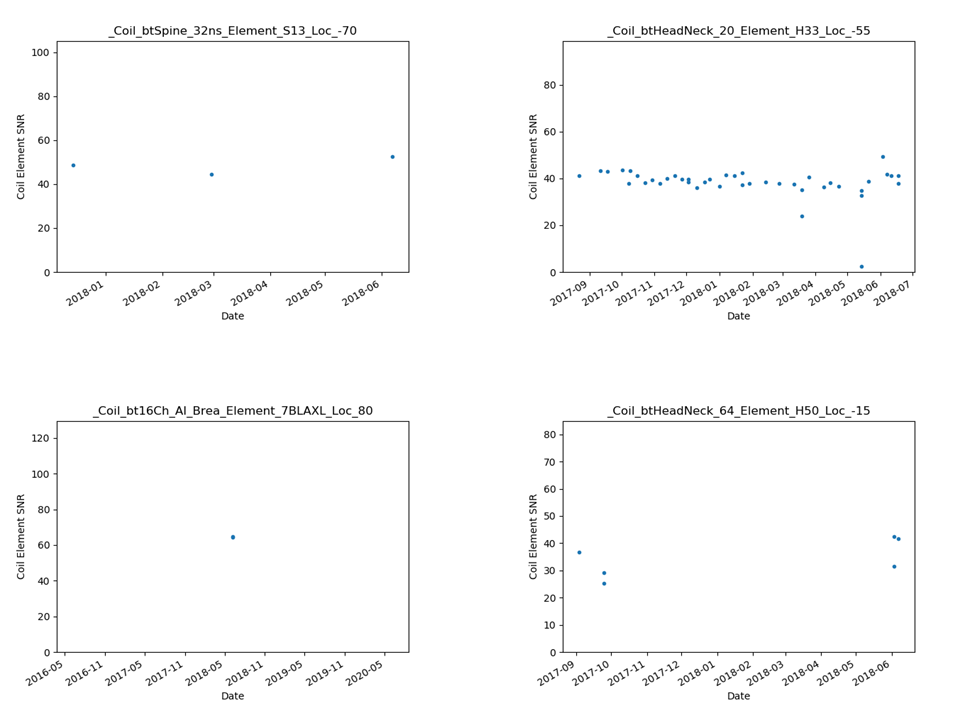
It was immediately notable that a very large number of images are generated by the coil-QA tests: since coil elements are distributed at a range of z-locations, it is necessary to generate images at multiple slice locations. However, the Siemens coil-QA test produces images for every coil element *at every tested slice location*, regardless of whether the coil element is located at this slice location or not. The result is that multiple images for each coil element are generated, one for each slice location (for this dataset, 1-12 slice locations are used depending on the coil) as well as summed images where the signal from each coil element has been combined.



*Figure 2: Coil-QA test hierarchy. The test data contained 11 different coils. One of these (HeadNeck\_20) has been fully illustrated, with 20 coil elements, each with images being produced at different slice locations.*

In total, the test dataset consisted of 5,566 DICOM images. All of these images were processed using the image analysis software described above, taking around 5-6 minutes to complete using a quad-core, 2.4GHz laptop.

Having processed the images and written the results into a csv file, the results analysis script was run. This script collated the results that have the same coil, coil element and test slice location, and then created and saved a plot of the coil-element SNR against time for each group. In total there were 886 unique groups



*Figure 3: Example plots output by the results analysis software. Each unique group has been given a name based on its coil, coil-element and slice location.*

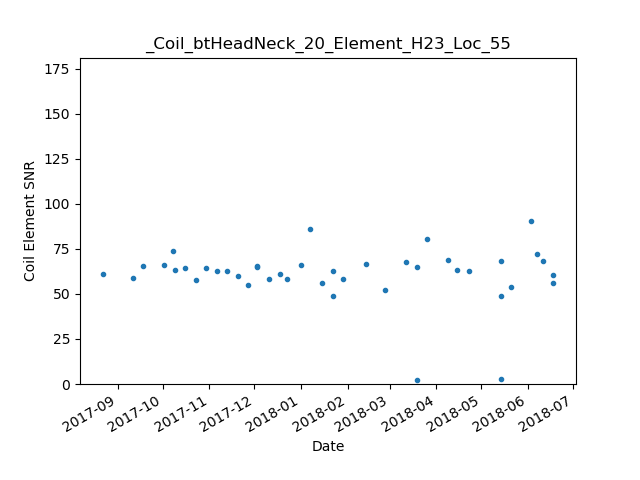
**Discussion and Recommendations for Future Work**

A couple of points can be immediately made about the data:

Firstly, on the whole, the data is quite sparse. Most of the coils were only tested on 1-3 dates during the 10-month period; only one coil (HeadNeck\_20) was tested routinely during this time.

Secondly, much of the data is redundant. Since an image is produced for each coil element for each slice location, a large fraction of the data is made up of images with a very low SNR. It should be possible to filter the data so that only the highest SNR slice locations are used for each element. For certain coil elements this isn't straightforward, as some coil elements were measured to have similar SNR values at two different slice locations.

For the HeadNeck\_20 coil, where sufficient data was available to assess the response of each element over time, it can be seen that the average SNR remains fairly constant (figure 4). The data is quite noisy however, with poor reproducibility between tests. As a result, only fairly gross changes in SNR will be detectible. It is not clear the method implemented in this work will pick up any system faults that are not already being detected by the manufacturer coil-QA. However, this work has enabled the longitudinal test results to be analysed, which might be beneficial in assessing long-term changes in system performance.



*Figure 4: SNR readings for element H23 of the HeadNeck\_20 coil.*

From the SNR data that has been generated during this work it is not possible to determine the causes of the variation in the data. The SNR of a single image produced by the coil-QA tests will be influenced by several factors, including the response of the coil element under test, the geometry of the test setup and the RF and magnetic behaviour of the overall system.

It would be an interesting extension to this work to try and assimilate the results from all of the tests, and use them to try and assess the overall behaviour of each coil, and also the behaviour of the system as a whole. It is possible that by doing so, a more sensitive measure of system stability might be achievable.

If the output of the scripts developed as part of this work prove to be useful, it would be possible to set up a DICOM node to automatically collect coil-QA images and run them through this analysis. By gathering data for multiple scanners over a long period of time, it might be possible to assess inter-scanner performance for similar units that are running the same tests.

The software produced for this work is specifically designed to review the images generated by the Siemens scanner from which the sample coil-QA data was taken. Any future effort to extend this work will be required to generalise the software to accept images from all of the MRI scanners used by the Trust.

**References**

|  |  |
| --- | --- |
| [1] | A. McCann, A. Workman and M. C, “A quick and robust method for measurement of signal-to-noise ratio in MRI,” *Physics in Medicine and Biology,* vol. 58, pp. 3775-3790, 2013. |
| [2] | G. Zack, W. Rogers and S. Latt, “Automatic measurement of sister chromatid exchange frequency,” *J. Histochem Cytochem,* vol. 25, no. 7, p. 74153, 1977. |