Parallel Imaging for Efficient Spike Noise Detection and Correction

F. Huang[1], W. Lin[1], Y. Li[1], and A. Reykowski[1]
[1]Invivo Corporation, Gainesville, Florida, United States

Introduction

Spike noise is a term used to describe broadband electrical interference in an MRI system. The result of spike noise can be seen in *k*-space as a bright dot. After the image reconstruction processing, the spike noise produces an undesired corduroy or wavy pattern over the magnetic resonance image. Typical sources of spike noise include unclosed door, a failed receiver channel, electrostatic discharge (ESD), and metal-on-metal vibration caused by loose mechanical fasteners or metal debris. Usually, the corrupted data has to be abandoned and the scan needs be repeated. Several methods have been proposed to remove spike noises using thresholds [1] or image space post processing [2] to avoid the repeated scan. As discussed in Ref [3], however, it is difficult to apply these techniques for some applications. In this work, a partially parallel imaging based technique, COnvolution and Combination OperAtion [4], is applied to efficiently detect and correct the spikes.

Theory

Parallel imaging methods based on *k*-space convolution [5, 6] work on the principle that the *k*-space signal can be approximated by a linear combination of signals from neighboring locations in multiple coil elements. This principle is adopted in COCOA for *k*-space error detection and correction with a fully-sampled multi-channel *k*-space data set. Convolution step: With a convolution kernel similar to the one in GRAPPA [3] (but more compact, as shown by Fig. 1), data convolution is performed to produce an extra synthetic *k*-space dataset with dispersed and reduced error. The convolution kernel is self-calibrated with the acquired data, which can be potentially corrupted. Combination step: By comparing the synthetic and the acquired data sets, the location with corrupted data can be identified. At locations with errors, the data from the synthetic *k*-space will be used. At other locations, the acquired *k*-space data will be used. To detect spikes, the magnitude of the difference between the synthetic and the original *k*-space can be used. At locations with spike, the difference will be much larger than the differences at other locations. Hence, spikes can be easily detected using histogram based method without using pre-defined threshold. If a spike is detected in the *k*-space region for convolution kernel calibration, the spike is removed from the calibration data and the convolution kernel is recalculated to produce the final corrected values.

Methods and Results

A set of brain data set, acquired on a 3.0T Achieva scanner (Philips, Best, Netherlands), using an 8-channel head coil (Invivo, Gainesville, FL), was used in the following two experiments. In both experiments, 10 spikes were artificially added at random locations. For the experiment shown in the second row of Fig. 2, one spike was located at the *k*-space center. COCOA was applied to the artificially corrupted data. The convolution kernel defined by Fig. 1 was calibrated with the corrupted data. The total processing time was about 3 and 6 seconds for these two experiments respectively. Figs. 2a~2c show the results when no spike was located at the *k*-space center. All spikes were detected and accurately corrected. The residual error in this case is negligible. Figs. 2d~2e show the results when one of the spikes was located at the *k*-space center. Due to the location of this spike, the accuracy of the convolution kernel was significantly reduced. However, COCOA could still detect all spikes. In order to further improve accuracy, the convolution kernel was recalculated after removing the detected spikes out of the calibration signal. As a result, the corrected data again had negligible error (Fig. 2e) in this case however at the cost of doubled processing time.

Discussions and Conclusion

COCOA can robustly detect random errors in *k*-space. Besides application on motion compensation [4], it can be applied to detect and correct other non-motion related artifacts, such as spike noise. The algorithm requires one or two calibration steps, depending on whether spike occurs outside the calibration region or not. The computation time of COCOA is the similar to GRAPPA, which is usually in seconds. In conclusion, spike noise can be reduced to a negligible level in a few seconds using COCOA.

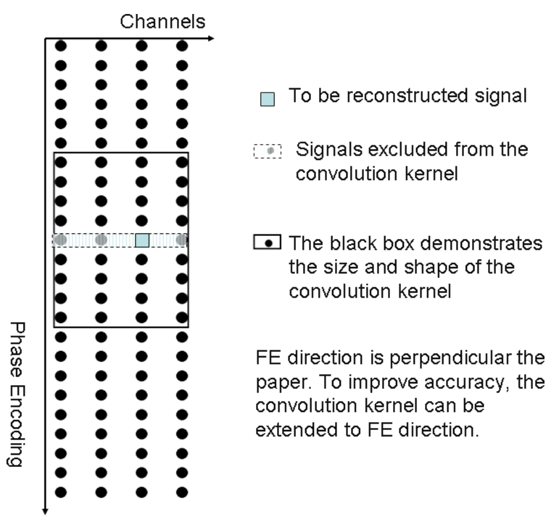

Fig. 1 Design of the Convolution kernel. Except for the shaded area, all of the source data points in the black box are used to reconstruct a separate copy of the signal in the colored box through convolution. FE denotes frequency encoding.

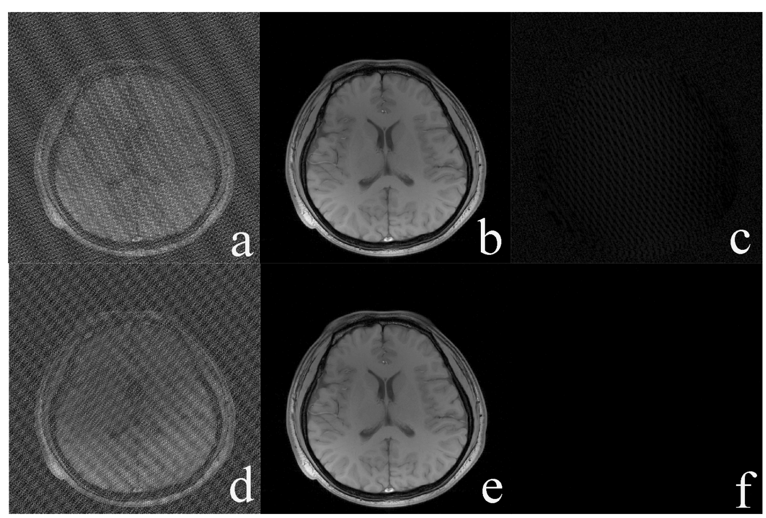

Fig. 2. Results of COCOA for data sets with 10 randomly located spikes. a) and d): reconstruction before COCOA. One spike is located at the center of *k*-space for d). b) and e): reconstruction after COCOA for a) and d) respectively. c) and f): the residual error of b) and e). They are brightened 100 times.

References [1] Staemmler M, et al. Magn Reson Med 1986;3:418–424 [2] Kao Y-H, et al. IEEE Trans Med Imaging 2000;19:671–680 [3] Chavez S, et al. Magn Reson Med 2009;62:510-519 [4]Huang F, et al. 2009; the 3rd International Workshop on Parallel MRI. Santa Cruz, CA, USA. [5] Yeh E. N., et. al. Magn Reson Med 2005;53:1383-1392 [6]Griswold M. A. , et. al. Magn Reson Med 2002;47:1202-1210