Published in final edited form as:

Proc SPIE Int Soc Opt Eng. 2012 February 12; 8319: 83190H-. doi:10.1117/12.910800.

Automating PACS Quality Control with the Vanderbilt Image Processing Enterprise Resource

Michael L. Esparza^{a,*}, E. Brian Welch^b, and Bennett A. Landman^{b,c,d}

^aComputer Science, Vanderbilt University, Nashville, TN, USA 37235

^bRadiology and Radiological Sciences, Vanderbilt University, Nashville, TN, USA 37235

^cElectrical Engineering, Vanderbilt University, Nashville, TN, USA 37235

^dBiomedical Engineering, Johns Hopkins University, Baltimore, MD 21218

Abstract

Precise image acquisition is an integral part of modern patient care and medical imaging research. Periodic quality control using standardized protocols and phantoms ensures that scanners are operating according to specifications, yet such procedures do not ensure that individual datasets are free from corruption—for example due to patient motion, transient interference, or physiological variability. If unacceptable artifacts are noticed during scanning, a technologist can repeat a procedure. Yet, substantial delays may be incurred if a problematic scan is not noticed until a radiologist reads the scans or an automated algorithm fails. Given scores of slices in typical three-dimensional scans and wide-variety of potential use cases, a technologist cannot practically be expected inspect all images. In large-scale research, automated pipeline systems have had great success in achieving high throughput. However, clinical and institutional workflows are largely based on DICOM and PACS technologies; these systems are not readily compatible with research systems due to security and privacy restrictions. Hence, quantitative quality control has been relegated to individual investigators and too often neglected. Herein, we propose a scalable system, the Vanderbilt Image Processing Enterprise Resource–VIPER, to integrate modular quality control and image analysis routines with a standard PACS configuration. This server unifies image processing routines across an institutional level and provides a simple interface so that investigators can collaborate to deploy new analysis technologies. VIPER integrates with high performance computing environments has successfully analyzed all standard scans from our institutional research center over the course of the last 18 months.

Keywords

Automated Image Processing; Quality Control; DICOM; PACS

*michael.a.esparza@vanderbilt.edu; <http://masi.vuse.vanderbilt.edu>; Medical-image Analysis and Statistical Interpretation Laboratory, Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, USA 37235.

1. INTRODUCTION

Medical imaging has provided powerful insights into understanding the structure and functional architecture of human anatomy and is widely used for the diagnosis, intervention, and management of clinical disorders. As we explore ever more subtle anatomical correlations in health and disease, we must look towards *efficiently* acquiring data on diverse subjects and making best use of this information. Quality assurance and verification of data integrity is a challenging problem in any image study. Manual inspection is resource intensive and difficult to quantify. In emerging population-level studies, it is impracticable to manually inspect tens of millions of images. The American College of Radiology published specific quality control standards for scanner calibration using specific phantom studies [2]. Yet, phantom-based certifications do not reflect on the suitability for a particular scan [1]. Data quality is intrinsically tied to research intent: data suitable for visual inspection for presence of lesions may be inappropriate for volumetric assessment.

Specific metrics, such as signal-to-noise ratio, variance of model residuals, and outlier detection, are well-understood to capture aspects of images that relate to the uncertainty with which one could infer a true (ideally acquired) value from that of a particular observation [3, 4]. Along with modern statistical resampling approaches (e.g., boot strap, jackknife, boosting/bagging), these approaches provide a systematic foundation for assessing the uncertainty in imaging data. Developing a metric to assess quality for particular kind of imaging data is a relatively straightforward process of selecting relevant metrics and monitoring for acceptable tolerances. However, these approaches remain *ad hoc* since selection of the specific criteria on which to base accept/reject quality decisions must be specified during the creation process. It remains an open question as to how to systematically answer the ubiquitous concerns, “Are these data useful?” and “Should I reacquire this dataset?”

A mechanism to measure quantitative quality in a general manner from the perspective of the applicability of the data would improve statistical power and validity of research studies and lead to more effective imaging for patient care. Herein, we propose a system to exploit widely available and industry standard picture archive and communication system (PACS) infrastructure to provide investigators, technologists, and clinicians rapid visual and quantitative feedback on the content and quality of their data.

2. METHODS

2.1 Server Overview

The proposed system, the Vanderbilt Image Processing Enterprise Resource–VIPER, operates on a model of ridiculously simple parallel processing (RSPP). RSPP involves division of a series of standalone programs onto multiple independent processors—no shared memory or inter-process communication is permitted while processes are running. RSPP stands in contrast to advanced parallel processing methods that seek fine grained control of computation (e.g., multi-core or GPU approaches), but offers a generic approach to parallelization that does not require domain specific knowledge. In our approach, processing consists of four steps: data aggregation, protocol matching, computation, and reporting

(Figure 1). The control system was implemented in Matlab (Mathworks, Natick, MA) with a MySQL (<http://www.mysql.com/>) database backend on a CentOS 64 bit Linux Server.

Data aggregation—Imaging data are aggregated from two sources. First, all data acquired at our imaging institute are automatically routed to a central PACS server. The PACS is running the open source DCM4CHEE software which supports both traditional DICOM access and web-access DICOM objects (WADO) calls [5]. VIPER monitors the PACS database for the arrival of new scans, and upon arrival, collects summary information regarding the protocol and unique Digital Imaging and Communications in Medicine (DICOM) identifiers for its local database.

Second, VIPER periodically monitors a shared file system for user deposited files. When new files are detected, filenames and protocol information are extracted for the database. This ensures that all scans acquired locally are detected and users have the flexibility deposit additional data without the need to allow users direct write access to the PACS.

To ensure that VIPER meets ensure ethical data use and patient privacy (including compliance with Health Insurance Portability and Accountability Act, HIPAA), the system runs as a service of the institute and only summary information on performance (as presented herein) are monitored. All datasets are stored, transmitted, and processed in anonymized form. When an investigator encounters a problem or question with a dataset, the investigator generates an acceptable query and/or sample image and presents the issue to institute staff. Essentially, the VIPER system is considered an extension of the scanning process.

Protocol matching—VIPER stores a list of available modules. Each module is associated with a list of keywords (i.e., “tags”). Periodically, the system probes all known files against keywords. A dataset with a keyword and a module form a job. If the job is not already recognized, a new job is added to the database.

Computation—VIPER uses a MOAB/TORQUE cluster to achieve high throughput. Jobs detected from the PACS database and shared file systems are forked to the cluster in batches. Periodically, VIPER probes for the number of running jobs. If it is below the maximum permitted, a job is retrieved from the queue. Datasets are either copied from the PACS via a Web Access to DICOM Objects (WADO) query or from the shared file system to higher performance cluster storage. VIPER creates a script for data analysis and submits it to the cluster.

Reporting—Periodically, VIPER monitors the cluster for completed jobs. When a job has finished, its status is evaluated. If the job was killed by system for failing to meet running time or memory constraints, the parameters are automatically doubled (one time only) and the job is put back in the queue. If the jobs failed, error status is indicated. If the job was successful, the summary report file (in PDF format) and any associated output datasets are copied a file store for retrieval by the PI.

Database—The organization of VIPER is rooted in a MySQL database with three key data tables for information processing and management. These tables are JobQueue, ProcessedResults, and SourceCache. These tables work in conjunction to store a job's information, current status, and final output directories.

The JobQueue table is the first part of the database a job encounters once it is located for processing. Each job that enters the queue is designated a unique ID number. A job is also assigned a processID number for the correct script association based on the type of job detected. Each job holds a status number in this table for monitoring. The job's exact status is known at all times and allows the server to act on a job if a status indicates its necessity.

The ProcessedResults table is where a processed job's general information is stored after completion. Various timestamps are created, runtimes are calculated, report files are created, etc. This table is vital for each server user.

The SourceCache table is used for specific information regarding each scan including scan timestamp and the type of scan. This is more of an internal use table for the server than the user.

2.2 Modules (Protocols)

Processing modules are defined by Matlab scripts. Each module must take two arguments, a file to process and a directory to which it is allowed to write. Each module must return a PDF file illustrating the results for quick visual assessment and may also return a Matlab datasstructure containing summary statistics. If summary statistics are returned, VIPER creates Shewhart plots based on these metrics for system administration and high level quality control [6]. Figure 2 illustrates PDF reports from modules implemented for functional MRI (fMRI), diffusion tensor imaging (DTI), chemical exchange saturation transfer (CEST) in the breast, and structural brain imaging using Freesurfer [7].

3. Discussion

We have reached a critical juncture: no longer can we rely on manual data inspection and fine tuning of parameters for individual datasets if we are going to meet the challenge of routinely interpreting millions of images. Human oversight will be required, but must be made with *just enough* input so as to ensure valid results and enable efficient supervision of large datasets. Much recent progress has been made in defining anatomical ontologies for clinical and imaging data (structured representations), data provenance (records of data flow and use), and schema (defining storage and relationships among data entities) [8–14]. Yet, numerous fundamental challenges remain in terms of validating analysis methods, integrating robust statistical techniques, and adopting industrial engineering approach for quality control (e.g., [15, 16]). Figure 3 illustrates that the VIPER system has successfully processed a large quantity of data. Users have anecdotally reported satisfaction with rapid, visual summaries of their data. Further, systematic optimization and evaluation of the proposed quality assurance framework is ongoing.

3.1 Technical Notes

VIPER is but one of many potential pipeline environments and distributed processing control systems that could be used. LONI [17], NAMIC [18], and JIST [19] provide RSPP access to high performance grid computing and could have been plausibly used with this system. Furthermore, XNAT [20] provides both a storage solution and a workflow management system. In fact, our institution is in the process of deploying a large-scale XNAT archive. Resource limitations and requirements for a stable operating procedure made such deployments infeasible.

We posit that a fundamental challenge confronting large-scale processing is not of algorithms or available technologies, but rather of a mismatch between expertise of stakeholders and the software engineering development practices. Imaging informatics infrastructure is based on standard DICOM services and objects, while leading edge analysis routines are based on research formats and multi-language scripts. The principle advantage of VIPER is that it can directly and simply exploit existing institutional infrastructure (standard PACS servers) and physicist provided data analysis scripts (typically written in Matlab or as combination of simple Unix command line calls). Since neither of these systems requires a change in the way these experts pursue their goals, the barriers to deployment are much lower than an integrated grid processing paradigm or a full XNAT system. Hence, VIPER allows investigators to experience the benefits of streamlined high performance computing without committing to particular development and/or analysis strategy.

3.2 Conclusion

The VIPER system provides a foundation for high-throughput quality control and interface with clinical systems using the DICOM standard. To illustrate the need for innovation in quality monitoring, compare the Shewhart chart for fMRI SNR from the Friedman-Glover manuscript with a Shewhart chart from the VIPER system in Figure 4. First, notice that the VIPER chart contains 10–30 datasets per day while the Friedman-Glover plot is sampled roughly once per week. Second, notice that the variance on the VIPER SNR is extreme — the six-standard deviation range (a typical outlier criterion) encompasses –10:1 to 110:1. Hence, the VIPER Shewhart chart, as presented, is not sensitive to problematic scans. This is because VIPER is designed (and is successful) at evaluating all images and does not currently monitor scan protocol or methods. Some scans included in this analysis are indeed nearly noise-only (e.g., during protocol development on research scanners at 7T) while others are of likely exceptional quality (e.g., phantom studies at 3T).

Through our efforts developing and implementing VIPER, we have found that computation time is vanishingly inexpensive. At the Vanderbilt Advanced Computing Center for Research and Education (ACCRE), the total cost for computation time (including amortized hardware purchase, maintenance, IT support staff, power, cooling, etc.) is *\$0.021 per hour*. MRI scan time runs a minimum of *\$8 per minute*, while minimum human wage is at least *\$11.16 per hour* (including indirect costs). Therefore, it is reasonable and prudent to spend hundreds of hours of computation to save a few minutes of scan or labor time. As clusters with thousands of processors are routinely available for parallel computation, many CPU

days of work may be accomplished in a manner of minutes. In our efforts, we advocate processing all possible datasets as quickly as possible and applying *post hoc* methods to identify when the processing was effective (as opposed to investigating predictive modeling to determine when a procedure might work). Even for standalone applications that must keep up with streaming data in real-time, vast computational resources are economically feasible.

In summary, the VIPER system serves an essential gateway platform to enable automatic, online quality assurance and data processing. These scripts are simple to develop, deploy, and optimize so that comparative studies of potential best practices in quality assurance may be conducted. Through these efforts, VIPER provides a framework for develop use cases, defining necessary system requirements, and advancing scientific goals while enterprise-scale integrated institutional commitments are explored (e.g., unified vendor neutral archives).

Acknowledgments

This project was supported by NIH 1R03EB012461, NIH N01-AG-4-0012, and the Vanderbilt Discovery program. *This work described herein has not been submitted elsewhere for publication or presentation.*

References

1. Friedman L, Glover GH. Report on a multicenter fMRI quality assurance protocol. *Journal of magnetic resonance imaging : JMRI*. 2006; 23(6):827–39. [PubMed: 16649196]
2. Chen CC, Wan YL, Wai YY, et al. Quality assurance of clinical MRI scanners using ACR MRI phantom: preliminary results. *Journal of digital imaging : the official journal of the Society for Computer Applications in Radiology*. 2004; 17(4):279–84.
3. Woodard JP, Carley-Spencer MP. No-Reference Image Quality Metrics for Structural MRI. *Neuroinformatics*. 2006; 4(20):243–262. [PubMed: 16943630]
4. Stocker T, Schneider F, Klein M, et al. Automated quality assurance routines for fMRI data applied to a multicenter study. *Human Brain Mapping*. 2005; 25(2):237–46. [PubMed: 15846770]
5. Warnock M, Toland C, Evans D, et al. Benefits of Using the DCM4CHE DICOM Archive. *Journal of Digital Imaging*. 2007; 20(0):125–129. [PubMed: 17917780]
6. Simmons A, Moore E, Williams SC. Quality control for functional magnetic resonance imaging using automated data analysis and Shewhart charting. *Magnetic resonance in medicine : official journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine*. 1999; 41(6):1274–8.
7. Dale AM, Fischl B, Sereno MI. Cortical surface-based analysis. I. Segmentation and surface reconstruction. *Neuroimage*. 1999; 9(2):179–94. [PubMed: 9931268]
8. Fenstermacher, D.; Street, C.; McSherry, T., et al. The Cancer Biomedical Informatics Grid (caBIG™). *Conf Proc IEEE Eng Med Biol Soc*; 2005. p. 743-6.
9. Kakazu KK, Cheung LW, Lynne W. The Cancer Biomedical Informatics Grid (caBIG): pioneering an expansive network of information and tools for collaborative cancer research. *Hawaii Med J*. 2004; 63(9):273–5. [PubMed: 15540527]
10. Kennedy D. Neuroinformatics and the Society for Neuroscience. *Neuroinformatics*. 2007; 5(3): 141–142.
11. Langella, S.; Oster, S.; Hastings, S., et al. The Cancer Biomedical Informatics Grid (caBIG) Security Infrastructure. *AMIA Annu Symp Proc*; 2007. p. 433-7.
12. Marcus DS, Olsen TR, Ramaratnam M, et al. The Extensible Neuroimaging Archive Toolkit: an informatics platform for managing, exploring, and sharing neuroimaging data. *Neuroinformatics*. 2007; 5(1):11–34. [PubMed: 17426351]

13. Toga AW, Crawford KL. The informatics core of the Alzheimer's Disease Neuroimaging Initiative. *Alzheimers Dement*. 2010; 6(3):247–56. [PubMed: 20451873]
14. von Eschenbach AC, Buetow K. Cancer Informatics Vision: caBIG. *Cancer Inform*. 2007; 2:22–4. [PubMed: 19458755]
15. Martone ME, Gupta A, Ellisman MH. E-neuroscience: challenges and triumphs in integrating distributed data from molecules to brains. *Nature neuroscience*. 2004; 7(5):467–72.
16. Akil H, Martone ME, Van Essen DC. Challenges and opportunities in mining neuroscience data. *Science*. 2011; 331(6018):708–12. [PubMed: 21311009]
17. Rex DE, Ma JQ, Toga AW. The LONI Pipeline Processing Environment. *Neuroimage*. 2003; 19(3):1033–48. [PubMed: 12880830]
18. Pieper, S.; Lorensen, B.; Schroeder, W., et al. The NA-MIC Kit: ITK, VTK, pipelines, grids and 3D slicer as an open platform for the medical image computing community. 3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro; 2006. p. 698-701.
19. Lucas BC, Bogovic JA, Carass A, et al. The Java Image Science Toolkit (JIST) for rapid prototyping and publishing of neuroimaging software. *Neuroinformatics*. 2010; 8(1):5–17. [PubMed: 20077162]
20. Marcus, DS.; Olsen, TR.; Ramaratnam, M., et al. XNAT: A Software Framework for Managing Neuroimaging Laboratory Data. Toronto, Ontario, Canada: 2005.

Figure 1.

VIPER constantly monitors incoming data on a PACS and an institute's file share (left); DICOM files are pulled from a research PACS via WADO. Protocols in these files are compared to a module list within VIPER and are distributed to the cluster for processing. The files are then collected and pushed back to a shared file server.

Figure 2. Example quality assurance PDF reports: (A) DTI, (B) Structural Brain, (C) fMRI, and (D) CEST. The PDF reports share a common format displaying subject information and acquisition details from the image file header in a text area (blurred above). Representative sections of the data are presented for qualitative interpretation. When multi-step processing is involved, visual representations of each step are presented to allow for qualitative quality control.

Server Load for Quality Assurance Modules

Module Name	Number of Datasets	Time (min) ($\mu + \sigma$)	Total time (days)
fMRI	4803	4.76 ± 6.73	15.5
DTI	800	2.45 ± 6.41	1.0
CEST	676	4.59 ± 16.9	2.1
TOTAL	6279	11.8 ± 30.0	18.6

Figure 3.

VIPER System Performance. A two week moving average (upper pane) shows the number of datasets in VIPER by acquisition day by module. The total number of datasets and processing times are shown in the lower pane.

Figure 4.

Comparison of Shewhart charts of fMRI SNR for of phantom data (A, Fig 13 of [1]) with empirical research data from VIPER (B). On both plots, each point represents one scan.