A Unified Pipeline for ECG Imaging Testing

Jess Tate[1], Eelco van Dam[2], Wilson Good[1], Jake Bergquist[1], Peter van Dam[3], Rob MacLeod[1]

[1]University of Utah, Salt Lake City, Utah, USA
[2]PEACS, The Netherlands
[3]UMC Utrecht, The Netherlands

Abstract

The Consortium for ECG Imaging (CEI) has formed several collaborative projects to evaluate and improve technical aspects of Electrocardiographic Imaging (ECGI), but these efforts are not yet implemented into an integrated software framework. We developed a framework to unify the multiple techniques and stages of ECGI into one pipeline. This framework merges existing open source packages: SCIRun, a problem solving environment; the Forward/Inverse toolkit, a series of SCIRun modules for ECGI; and PFEIFER, a cardiac signal pre-processing tool. The Unified ECGI Toolkit (UETK), combined with the EDGAR dataset, allows users to test and validate a vast array of parameters within each stage of the ECGI pipeline. We expect that this unified tool will help introduce new researchers to ECGI, facilitate interaction between the various groups working on ECGI, and establish a common approach for researchers to test and validate their ECGI techniques.

1. Introduction

Electrocardiographic imaging (ECGI) is the process of using patient geometry and body surface recordings to calculate high resolution cardiac activity to facilitate noninvasive arrhythmia detection and treatment. Although many ECGI methods have shown the potential to improve clinical outcomes, their adoption into clinical practice has been hindered by technical challenges.

The Consortium for ECG Imaging (CEI) [1] was formed to leverage collaborative efforts toward overcoming technical challenges associated with ECGI. By focusing the efforts of multiple research groups around the world, the CEI has formed collaborative projects to evaluate and improve many of the technical aspects of ECGI, including: generating forward models [2–6], pre-processing body surface recordings [7–9], inverse calculation method [2, 10], activation and recovery time determination, and others. Each of these groups has made progress to improve ECGI, yet these efforts are still part of many disparate software pipelines, many of which are not available.

We developed a framework to unify multiple techniques and stages of ECGI into the most comprehensive open source pipeline to date. This framework merges existing open source packages: SCIRun [11], the Forward/Inverse toolkit [12,13], and PFEIFER [14]. The goal of this unified tool, combined with the EDGAR [15] dataset, is to provide a single platform which can be used to perform a plurality of ECGI tasks without switching software. The resulting Unified ECGI Toolkit (UETK, `https://github.com/CECGImaging/UETK`) can be used by many ECGI researchers, notably: those new to field, in collaborative projects who wish to share techniques, and who need to test and validate innovative techniques.

2. Methods

The unified tool that we developed for ECGI combines the functionality of existing complementary open source packages form a more complete yet still generalizable pipeline. We combined general (SCIRun) and specific tools (Forward/Inverse Toolkit and PFEIFER) to simplify some of the of cumbersome aspects of ECGI while maintaining the flexibility to handle a variety of tasks and data.

SCIRun (`https://scirun.org`) [11] is a modular based problems solving environment and is used as the base of the UETK. SCIRun provides a flexible, yet consistent data environment that is well suited for geometry based modeling such as ECGI. SCIRun's rendering tools also help to quickly visualize result of any of the stages of the pipeline, and its modular structure supports a array of geometry processing tools, such as: registration, data mapping, interpolation, simulation and others. One of pieces that is key to enabling this UETK pipeline is SCIRun's interface with scripting software: Python and MATLAB. Using these scripting languages within SCIRun exposes a host of possibilities and allows us to leverage existing software.

The Forward/Inverse Toolkit [12, 13] (`https://github.com/SCIInstitute/FwdInvToolkit`) is set of modules and examples within SCIRun designed to
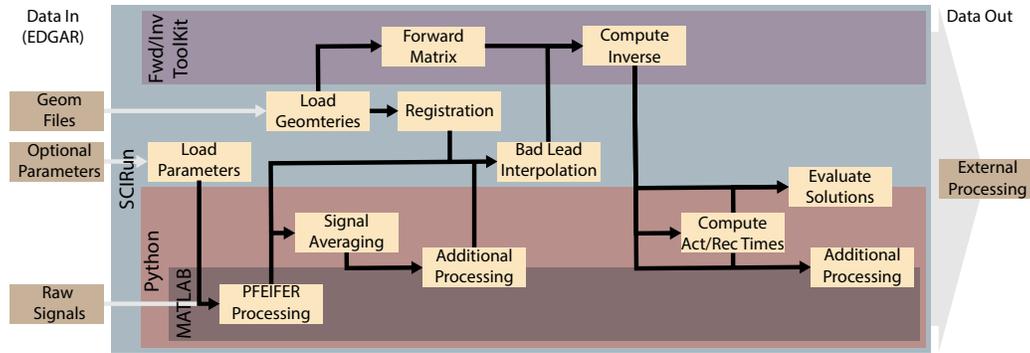
Figure 1. Pipeline for the unified ECGI toolkit (UETK). The flexible pipeline utilizes SCIRun's interface with Python to interact with MATLAB and PFEIFER and to perform custom tasks. The pipeline can be simplified or expanded as needed.

perform ECGI tasks. The Forward/Inverse Toolkit includes methods for computing ECG inverse calculations: Tikhonov regularization, activation-based Gauss-Newton optimization, total variation, method of fundamental solutions, and others.

PFEIFER [14] is an interactive signal processing tool that is designed for multichannel cardiac potential recordings. In addition to baseline correction and filtering, PFEIFER is designed to facilitate beat extraction and choosing fiducial markers.

Figure 1 illustrates how each of these software interact within the UETK. SCIRun uses its interface with Python to initialize a MATLAB engine, which is needed to launch and run PFEIFER. Once it is finished, the isolated body surface recordings are further processed with signal averaging or additional filtering. The bad leads identified in PFEIFER are passed to SCIRun for interpolation. Meanwhile, the geometry data is loaded into SCIRun natively, from which the bad lead interpolation, registration, or other geometry processes are performed. Using the geometry as inputs to the Forward/Inverse Toolkit, SCIRun then calculates the forward matrix, if needed. The cardiac sources can then be calculated from the processed time signals using one of the many implemented inverse techniques. The cardiac signals are then evaluated and postprocessed using SCIRun's linear algebra modules, or with Python or MATLAB with the interface with Python.

The framework is designed to work with data in the EDGAR format [15], therefore we tested the pipeline on datasets found in the EDGAR dataset (`https:/edgar.sci.utah.edu`. As an example, we tested the effect of signal averaging on ECGI using the Nijmegen-2004-12-09 dataset. We used PFEIFER within the unified framework to auto segment multiple beats within a recording, then averaged each time point over all the beats selected. An additional lowpass filter (Butterworth, order 6, cutoff frequency 100 Hz) was applied after signal averaging. The ECGI was performed with BEM and Tikhonov regulariza-

tion. We compare ECGI solutions from a single extracted beat to solutions from a signal averaged beat.

3. Results

The results presented in this paper include the Unified ECGI Toolkit (UETK, `https://github.com/CECGImaging/UETK`) itself and the proof of concept established with the example data from EDGAR.

Figure 2 shows the UETK in use with the Nijmegen dataset. The pipeline contains all the computation pieces required to use the data from the EDGAR database and perform ECGI with multiple pre-processing options, including signal averaging and bad lead removal. The visualization tools in SCIRun provide visualization of the computed extracellular potentials on the heart surface and at any stage in the pipeline.

The effect of signal averaging on the body surface potentials are shown in Figure 3. The signal averaging noticeably smoothes the time signals, yet the low pass filter did not provide any noticeable additional smoothing. The effect of using these smoothed signals in ECGI is show in Figure 4. ECGI with smoothed body surface potentials produce cardiac signals that are smoothed to a qualitatively similar level. The difference of the signal using signal averaging and low pass filtering was approximately 5% of the magnitude of the signal. The smoothed signal produced minimal difference in the spatial distribution of the cardiac potentials compared to the solution without smoothed signals (not shown).

4. Discussion and Conclusions

Our Unified ECGI Toolkit (UETK), combined with the EDGAR dataset, allows users to test and validate a vast array of parameters within each stage of the ECGI pipeline. Users can input geometry and bioelectric signals, preprocess the data, compute the ECGI, evaluate the results,
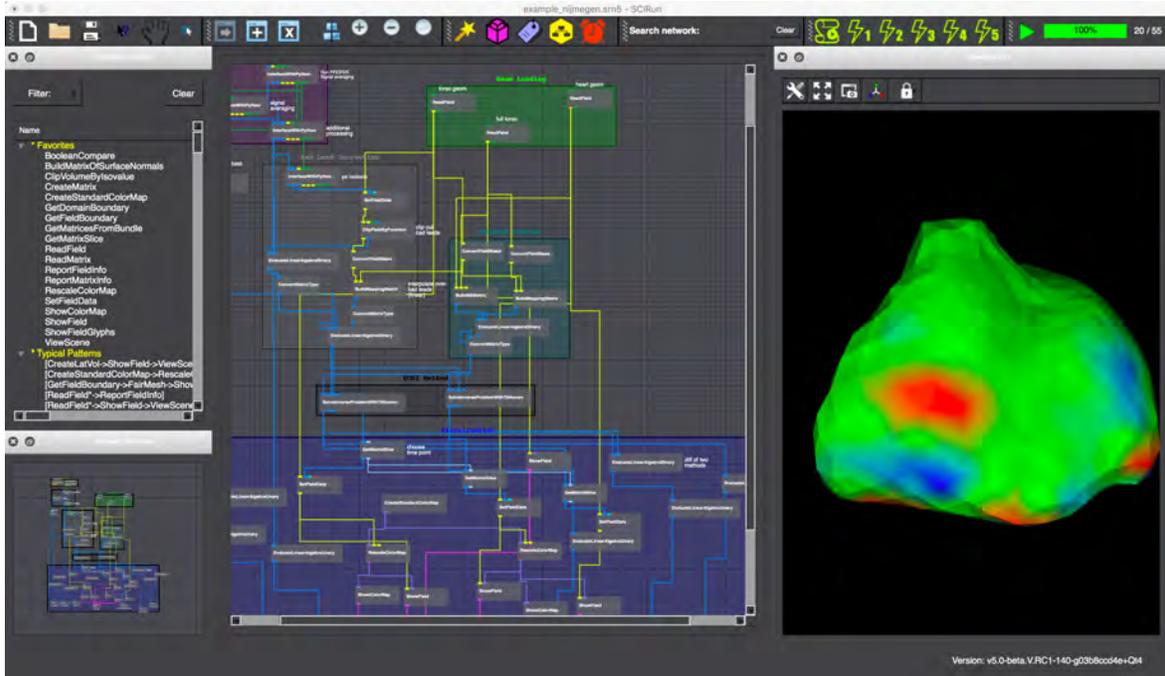
Figure 2. Example of the unified ECGI toolkit (UETK). Shown is the pipeline to extract ECG beats, signal averaging, low pass filtering, and a comparison of the original and processed beats and the associated ECGI solutions. The colors in the network demarcate the different computational pieces of the pipeline.
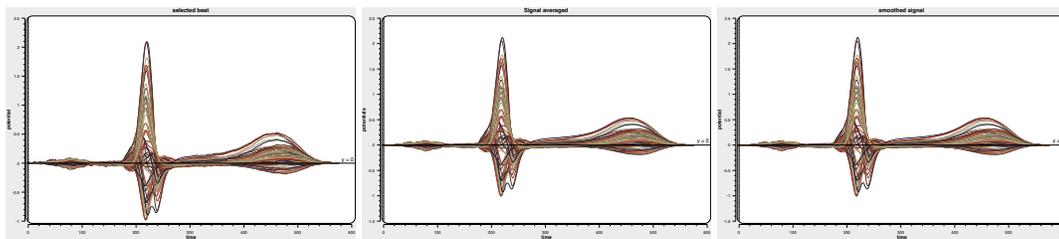

Figure 3. ECG tracings at various stages of preprocessing. From left to right: selected beat from PFEIFER, signal averaged, additional low pass filtering.
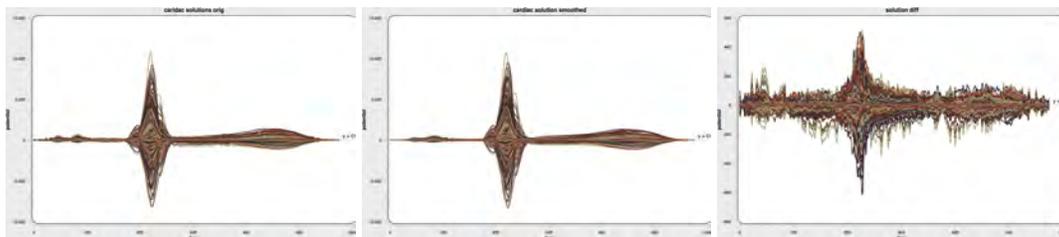
Figure 4. Compute cardiac signals using ECG signals from PFEIFER compared to signal averaged and filtered signals. From left to right: cardiac solutions using a single beat from PFEIFER, cardiac solutions using signal averaged beat, and the difference of two solutions.

and visualize all stages in one pipeline. Users can also modify each aspect of the pipeline, for instance, PFEIFER can be used to test the roles of beat selection and filter choice in the ECGI pipeline. Users can also replace each stage with custom MATLAB or Python methods through SCIRun's scripting interfaces.

The UETK unites many of pieces in ECGI, which have typically been performed by separate tools, into the most

comprehensive open source ECGI tool currently available. We expect that this unified tool will help introduce researchers to ECGI, facilitate interaction between the various groups working on ECGI, and establish a common approach for researchers to test and validate their ECGI techniques. The UETK can be found at `https://github.com/CECGImaging/UETK`.

Acknowledgments

References

[1] Coll-Font J, Dhamala J, Potyagaylo D, Schulze W, Tate J, Guillem M, van Dam P, Dossel O, Brooks D, MacLeod R. The Consortium for Electrocardiographic Imaging. In IEEE Computers in Cardiology, volume 43. 2016; .

[2] Ghimire S, Dhamala J, Coll-Font J, Tate JD, Guillem MS, Brooks BH, MacLeod RS, Wang L. Overcoming barriers to quantification and comparison of electrocardiographic imaging methods: A community- based approach. In Computing in Cardiology Conference (CinC), 2017, volume 44. 2017; 1–4.

[3] Tate J, Gillette K, Burton B, Good W, Zenger B, Coll-Font J, Brooks D, MacLeod R. Reducing error in ECG forward simulations with improved source sampling. Front Physiol 2018;9:1304.

[4] Tate JD, Zemzemi N, Good WW, van Dam P, Brooks DH, MacLeod RS. Effect of segmentation variation on ECG imaging. In Computing in Cardiology, volume 45. September 2018; .

[5] Tate J, Schuler S, Dössel O, MacLeod R, Oostendorp TF. Correcting undersampled cardiac sources in equivalent double layer forward simulations. In Functional Imaging and Modeling of the Heart. Bordeaux, France: Springer-Verlag, 2019; 147–155.

[6] Schuler S, Tate J, Oostendorp T, MacLeod R, Dössel O. Spatial downsampling of surface sources in the forward problem of electrocardiography. In Functional Imaging and Modeling of the Heart. Bordeaux, France: Springer-Verlag, 2019; 29–36.

[7] Bear L, Dogrusoz YS, Svehlikova J, Coll-Font J, Good W, van Dam E, MacLeod R, Abell E, Walton R, Coronel R, Haissaguerre M, Dubois R. Effects of ecg signal processing on the inverse problem of electrocardiography. In Computing in Cardiology Conference (CinC), 2018, volume 45. 2018; 1–4.

[8] Dogrusoz YS, Bear L, Svehlikova J, Good W, Coll-Font J, van Dam E, Dubois R, MacLeod R. Reduction of effects of noise on the inverse problem of electrocardiography with bayesian estimation. In Computing in Cardiology Conference (CinC), 2018, volume 45. 2018; 1–4.

[9] Svehlikova J, Zelinka J, Dogrusoz YS, Good W, Tysler M, Bear L. Impact of signal preprocessing on the inverse localization of the origin of ventricular pacing. In Computing in Cardiology Conference (CinC), 2018, volume 45. 2018; 1–4.

[10] Coll-Font J, Brooks DH. Tracking the position of the heart from body surface potential maps and electrograms. Front Physiol 2018;9:1727.

[11] Parker S, Johnson C. SCIRun: a scientific programming environment for computational steering. In Proc ACM IEEE Supercomputing Conf, volume 2. IEEE, Los Alamitos, CA, 1995; 1419–1439.

[12] Coll-Font J, Burton B, Tate J, Erem B, Swenson D, Wang D, Brooks D, van Dam P, MacLeod R. New additions to the Toolkit for Forward/Inverse Problems in electrocardiography within the SCIRun problem solving environment. In IEEE Computers in Cardiology, volume 40. 2014; 213–216.

[13] Burton B, Tate J, Erem B, Swenson D, Wang D, Brooks D, van Dam P, MacLeod R. A toolkit for forward/inverse problems in electrocardiography within the scirun problem solving environment. In Proceedings of the IEEE Engineering in Medicine and Biology Society 33rd Annual International Conference. IEEE Eng. in Med. and Biol. Soc., 2011; 1–4.

[14] Rodenhauser A, Good W, Zenger B, Tate J, Aras K, Burton B, MacLeod R. PFEIFER: Preprocessing framework for electrograms intermittently fiducialized from experimental recordings. J Open Source Software 2018;3(21):472.

[15] Aras K, Good W, Tate J, Burton B, Brooks D, Coll-Font J, Doessel O, Schulze W, Patyogaylo D, Wang L, van Dam P, MacLeod R. Experimental data and geometric analysis repository: EDGAR. J Electrocardiol 2015;48(6):975–981.

Address for correspondence:

Jess Tate
Scientific Computing and Imaging Institute
WEB, 72 South Campus Drive, Rm 3750
Salt Lake City, UT 84112
jess@sci.utah.edu