A brief history of the Gamera document image analysis system*

Michael Droettboom
Space Telescope Science Institute
3700 San Martin Drive, Baltimore, Maryland, USA

Gamera [1] arose out of an experience designing an optical music recognition (OMR) system: software that is able to “read” sheet music and convert it into formats for searching and listening. I, along with my colleague Karl MacMillan, began working on novel algorithms for this problem in 2000 as part of a project to digitize a major collection of United States of America historical sheet music [2], supported in part by the National Science Foundation (USA). As a starting point, we had inherited a large body of purpose-written source code from our thesis adviser, Ichiro Fujinaga [3]. While the software had impressive capabilities, it was difficult to improve. Different parts of the system were interdependent in confusing and subtle ways. At the same time, as we began to read the document analysis literature for new algorithms that might be applicable to our own problems, we discovered that the same methods were being used in all kinds of diverse domains, but there was very little software reuse between them. Frustration with these experiences, and a certain amount of *naïveté*, lead us to create a framework made up of reusable components narrowly targeted at building document analysis systems. With such a framework, document analysis researchers and computational humanists could focus on the problems specific to their domains, and spend less time reimplementing the underlying commodity pieces.

My past experience with the Python programming language[1] had proven it to be very easy-to-learn and well-suited to rapid and experimental programming. This made it an obvious candidate for the glue that would hold these components together. For speed reasons, most of the low-level and algorithmic parts of Gamera are implemented in C++. Gamera also builds on top of a number of other open-source projects, most notably Vision with Generic Algorithms (VIGRA) for low-level image processing [4], Genetic Algorithms Library (GALib)

*Published in C. Dalitz (Ed.): “Document Image Analysis with the Gamera Framework.” Schriftenreihe des Fachbereichs Elektrotechnik und Informatik, Hochschule Niederrhein, vol. 8, pp. 1-4, Shaker Verlag (2009)

[1] For more information on Python, see `http://python.org/`

for classifier optimization [5] and wxPython for the graphical user interface [6].

Aside from its original application to common music notation, Gamera was used internally at Johns Hopkins University for medieval manuscript (Roman de la Rose [7]), early Scottish census data [8], and the center labels of LP recordings [9]. Outside of the original core team, it has been used for recognition of the Navajo language [10], Early Latin texts [11], German lute tablature [12] and Psaltic Byzantine Chant Notation [13]. It has also been used as a framework in which new fundamental algorithms are developed [14] [15] and as the basis for a course in document image recognition [16]. It is important to note that Gamera is merely the underlying framework for the above projects. By writing new domain-specific functionality grouped into toolkits, in many cases, these projects were able to build end-to-end recognition systems.

From the beginning, Gamera was licensed under the GNU General Public License (GPL), version 2.0, [17] and its source code was freely available online. The GPL was important to us so that changes to the system are required to remain open source. This has been important to encourage international collaboration, since contributors know their work will remain open. Additionally, the source code itself is a vital and precise way to verify and reproduce research findings.

In January 2004, Gamera was posted to the software collaboration website SourceForge[2] to garner wider community involvement. SourceForge provided a source code repository, allowing developers outside of our in-house team to contribute to the project. Through this site, a number of third-party developers joined the team, along with a legion of beta testers and bug reporters.

Perhaps the most important benefit of open source has been the project’s longevity. When I left Johns Hopkins University and the field of document image recognition in 2005, it was reasonably easy for another developer from another institution, Christoph Dalitz from the Hochschule Niederrhein (Germany), to successfully take over management of the code. This would be much more difficult if the source code were closed. It is very rewarding to watch Gamera’s continued progress even though I no longer have time to actively participate.

[2] The project page of Gamera at SourceForge is `http://sourceforge.net/projects/gamera/`.

References

[1] M. Droettboom, K. MacMillan, I. Fujinaga: *The Gamera framework for building custom recognition systems.* Proceedings of the Symposium on Document Image Understanding Technologies, pp. 275-286 (2003) see also `http://gamera.informatik.hsnr.de/`

[2] L. S. Levy: *Guide to the Lester S. Levy Collection of Sheet Music.* `http://levysheetmusic.mse.jhu.edu/guide/` (1984)

[3] I. Fujinaga: *Adaptive optical music recognition.* Ph. D. Dissertation. McGill University, Montréal, Canada. (1997)

[4] U. Köthe: *Generic Programming for Computer Vision: The VIGRA Computer Vision Library* `http://hci.iwr.uni-heidelberg.de/vigra/` (2000-)

[5] M. Wall: *GAlib: A C++ Library of Genetic Algorithm Components* `http://lancet.mit.edu/ga/` (1999-)

[6] R. Dunn, N. Rappin: *wxPython in Action.* Manning. (2006) see also `http://wxpython.org/`

[7] S. G. Nichols, G. S. Choudhury: *Roman de la Rose Digital Library.* `http://romandelarose.org/#project` (1996-)

[8] M. Droettboom: *Correcting broken characters in the recognition of historical printed documents.* Proceedings of Joint Conference on Digital Libraries (JCDL), pp. 364-366 (2003)

[9] C. Lai, I. Fujinaga, C. Leive: *The challenges in developing digital collections of phonograph records.* Proceedings of Joint Conference on Digital Libraries (JCDL), pp. 332-333 (2005)

[10] K. Canfield: *A Pilot Study for a Navajo Textbase.* Proceedings of The 17th International Conference on Humanities Computing and Digital Scholarship (ACH/ALLC), pp. 28-30 (2005)

[11] S. Reddy, G. Crane: *A Document Recognition System for Early Modern Latin.* Chicago Colloquium on Digital Humanities and Computer Science (2006)

[12] C. Dalitz, C. Pranzas: *German Lute Tablature Recognition.* Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR), pp. 371-375 (2009)

[13] C. Dalitz, G. K. Michalakis, C. Pranzas: *Optical Recognition of Psaltic Byzantine Chant Notation.* International Journal of Document Analysis and Recognition 11, pp. 143-158 (2008)

[14] J. Hopkins, T. Anderson: *A Fourier-descriptor-based character recognition engine implemented under the Gamera open-source document-processing framework.* SPIE Proceedings, vol. 5676 (Document Recognition and Retrieval XII), pp. 111-118 (2005)

[15] A. Capela, A. Rebelo, J.S. Cardoso, C. Guedes: *Staff Line Detection and Removal with Stable Paths.* International Conference on Signal Processing and Multimedia Applications (SIGMAP), pp. 263-270 (2008)

[16] C. Dalitz: *Document Image Analysis.* Lecture at the Hochschule Niederrhein, Krefeld, Germany (summer terms 2008 and 2009)

[17] Free Software Foundation. *GNU General Public License, Version 2.* `http://www.gnu.org/licenses/gpl-2.0.html` (1991)