From Eigen

Eigen is a C++ template library for linear algebra: matrices, vectors, numerical solvers, and related algorithms.

Contents

Announcements

Eigen 3.3.7 released!

(11.12.2018)

Eigen 3.3.6 released!

(10.12.2018)

Eigen 3.3.5 released!

(23.07.2018)

Eigen now offers a git mirror on github!

(29.12.2017)

Get it

The **latest stable release** is Eigen 3.3.7. Get it here: tar.bz2 (http://bitbucket.org/eigen/eigen/get/3.3.7.tar.bz2), tar.gz (http://bitbucket.org/eigen/eigen/get/3.3.7.tar.gz), zip (http://bitbucket.org/eigen/eigen/get/3.3.7.zip). Changelog.

The **latest 3.2.10 (previous series) release** is Eigen 3.2.10. Get it here: tar.bz2 (http://bitbucket.org/eigen/eigen/get/3.2.10.tar.bz2), tar.gz (http://bitbucket.org/eigen/eigen/get/3.2.10.tar.gz), zip (http://bitbucket.org/eigen/eigen/get/3.2.10.zip). Changelog.

The **unstable** source code from the **development branch** is there: tar.bz2 (http://bitbucket.org/eigen/eigen/get/default.tar.bz2), tar.gz (http://bitbucket.org/eigen/eigen/get/default.tar.gz), zip (http://bitbucket.org/eigen/eigen/get/default.zip).

To check out the Eigen repository using Mercurial (http://www.selenic.com/mercurial/wiki/), also known as "hg", do:

```
hg clone https://bitbucket.org/eigen/eigen/
```

Eigen 3.3.4 released!

(15.06.2017)

There is also an official git mirror on github: https://github.com/eigenteam/eigen-git-mirror.

[other downloads (https://bitbucket.org/eigen/eigen/downloads/)] [browse the source code (https://bitbucket.org/eigen/eigen/src/)]

Overview

- **Eigen is versatile.**
  - It supports all matrix sizes, from small fixed-size matrices to arbitrarily large dense matrices, and even sparse matrices.
  - It supports all standard numeric types, including std::complex, integers, and is easily extensible to custom numeric types (http://eigen.tuxfamily.org/dox/TopicCustomizingEigen.html#CustomScalarType).
  - It supports various matrix decompositions (http://eigen.tuxfamily.org/dox/group__TopicLinearAlgebraDecompositions.html) and geometry features (http://eigen.tuxfamily.org/dox/group__TutorialGeometry.html).
  - Its ecosystem of unsupported modules (http://eigen.tuxfamily.org/dox/unsupported/index.html) provides many specialized features such as non-linear optimization, matrix functions, a polynomial solver, FFT, and much more.
- **Eigen is fast.**
  - Expression templates allow to intelligently remove temporaries and enable lazy evaluation (http://eigen.tuxfamily.org/dox/TopicLazyEvaluation.html), when that is appropriate.
  - Explicit vectorization is performed for SSE 2/3/4, AVX, AVX2, FMA, AVX512, ARM NEON (32-bit and 64-bit), PowerPC AltiVec/VSX (32-bit and 64-bit), ZVector (s390x/zEC13) SIMD instruction sets, and since 3.4 MIPS MSA with graceful fallback to non-vectorized code.
  - Fixed-size matrices are fully optimized: dynamic memory allocation is avoided, and the loops are unrolled when that makes sense.
  - For large matrices, special attention is paid to cache-friendliness.
- **Eigen is reliable.**
  - Algorithms are carefully selected for reliability. Reliability trade-offs are clearly documented (http://eigen.tuxfamily.org/dox/group__TopicLinearAlgebraDecompositions.html) and extremely (http://eigen.tuxfamily.org/dox/classEigen_1_1JacobiSVD.html) safe (http://eigen.tuxfamily.org/dox/classEigen_1_1FullPivHouseholderQR.html) decompositions (http://eigen.tuxfamily.org/dox/classEigen_1_1FullPivLU.html) are available.
  - Eigen is thoroughly tested through its own test suite (over 500 executables), the standard BLAS test suite, and parts of the LAPACK test suite.
- **Eigen is elegant.**
  - The API is extremely clean and expressive while feeling natural to C++ programmers, thanks to expression templates.
  - Implementing an algorithm on top of Eigen feels like just copying pseudocode.
- **Eigen has good compiler support** as we run our test suite against many compilers to guarantee reliability and work around any compiler bugs. Eigen also is standard C++98 and maintains very reasonable compilation times.

Documentation

- **Eigen 3 documentation (http://eigen.tuxfamily.org/dox)**: this includes a getting started guide (http://eigen.tuxfamily.org/dox/GettingStarted.html), a long tutorial (http://eigen.tuxfamily.org/dox/group__TutorialMatrixClass.html), a quick reference (http://eigen.tuxfamily.org/dox/group__QuickRefPage.html), and page about porting from Eigen 2 to Eigen 3 (http://eigen.tuxfamily.org/dox/Eigen2ToEigen3.html).

- **Eigen development branch documentation (http://eigen.tuxfamily.org/dox-devel)**

- **Eigen 2 documentation (http://eigen.tuxfamily.org/dox-2.0)** (old): this includes the Eigen 2 Tutorial (http://eigen.tuxfamily.org/dox-2.0/TutorialCore.html).

- FAQ (Frequently asked questions)

- Presentations and talks

Requirements

Eigen doesn't have any dependencies other than the C++ standard library.

We use the CMake build system, but only to build the documentation and unit-tests, and to automate installation. If you just want to use Eigen, you can use the header files right away. There is no binary library to link to, and no configured header file. Eigen is a pure template library defined in the headers.

License

Eigen is Free Software (http://www.gnu.org/philosophy/free-sw.html). Starting from the 3.1.1 version, it is licensed under the MPL2 (http://www.mozilla.org/MPL/2.0), which is a simple weak copyleft license. Common questions about the MPL2 are answered in the official MPL2 FAQ (http://www.mozilla.org/MPL/2.0/FAQ.html).

Earlier versions were licensed under the LGPL3+.

Note that currently, a few features rely on third-party code licensed under the LGPL: SimplicialCholesky, AMD ordering, and constrained_cg. Such features can be explicitly disabled by compiling with the EIGEN_MPL2_ONLY preprocessor symbol defined. Furthermore, Eigen provides interface classes for various third-party libraries (usually recognizable by the <Eigen/*Support> header name). Of course you have to mind the license of the so-included library when using them.

Virtually any software may use Eigen. For example, closed-source software may use Eigen without having to disclose its own source code. Many proprietary and closed-source software projects are using Eigen right now, as well as many BSD-licensed projects.

See the **MPL2 FAQ (http://www.mozilla.org/MPL/2.0/FAQ.html)** for more information, and do not hesitate to contact us if you have any questions.

Compiler support

Eigen is standard C++98 and so should theoretically be compatible with any compliant compiler. Whenever we use some non-standard feature, that is optional and can be disabled.

Eigen is being successfully used with the following compilers:

- GCC (http://gcc.gnu.org), version 4.8 and newer. Older versions of gcc might work as well but they are not tested anymore.
- MSVC (http://en.wikipedia.org/wiki/Visual_C%2B%2B) (Visual Studio), 2012 and newer. Be aware that enabling IntelliSense (/FR flag) is known to trigger some internal compilation errors. The old 3.2 version of Eigen supports MSVC 2010, and the 3.1 version supports MSVC 2008.
- Intel C++ compiler (http://en.wikipedia.org/wiki/Intel_C%2B%2B_Compiler). Enabling the `-inline-forceinline` option is highly recommended.
- LLVM/CLang++ (http://clang.llvm.org/cxx_status.html), version 3.4 and newer. (The 2.8 version used to work fine, but it is not tested with up-to-date versions of Eigen)
- XCode 7 and newer. Based on LLVM/CLang.
- MinGW (http://en.wikipedia.org/wiki/Mingw), recent versions. Based on GCC.
- QNX's QCC compiler.

Regarding performance, Eigen performs best with compilers based on GCC or LLVM/Clang. See this page for some known compilation issues.

Get support

Need help using Eigen? Try this:

- To get help, stackoverflow (http://stackoverflow.com/questions/tagged/eigen) is your best resource. You can also search within the forum archives (http://forum.kde.org/viewforum.php?f=74).
- You can always try our IRC channel.
- Want to discuss something with the developers? Use our mailing list.

Bug reports

For bug reports and feature requests, please use the issue tracker (http://eigen.tuxfamily.org/bz). To file a new bug, go there (http://eigen.tuxfamily.org/bz/enter_bug.cgi). See this page for some instructions.

Mailing list

Address: **eigen@lists.tuxfamily.org**

- To subscribe, send a mail with subject *subscribe* to eigen-request@lists.tuxfamily.org
- To unsubscribe, send a mail with subject *unsubscribe* to eigen-request@lists.tuxfamily.org

The Eigen mailing list can be used for discussing general Eigen development topics. End-user questions are often better asked on the Users Forum (http://forum.kde.org/viewforum.php?f=74). Development of specific features is best tracked and discussed on our Bugzilla (http://eigen.tuxfamily.org/bz). See this page.

This mailing list is **public** and has **public archives (http://listengine.tuxfamily.org/lists.tuxfamily.org/eigen/)**. The archives are mirrored by gmane.org (http://dir.gmane.org/gmane.comp.lib.eigen).

Important: You must subscribe before you may post. Sorry, this is our only way to prevent spam.

Important: After you sent your subscription request, you will receive a confirmation e-mail. Check your spam folder, as these confirmation e-mails are often filtered as spam!

There is also a **private** mailing list which should only be used if you want to write privately to a few core developers (it is read by Benoît, Gaël, Jitse and Christoph). The address is **eigen-core-team** at the same lists server as for the Eigen mailing list. You do not need to subscribe (actually, subscription is closed). For all Eigen development discussion, use the public mailing list or Bugzilla instead.

IRC Channel

Our IRC (http://en.wikipedia.org/wiki/Internet_Relay_Chat) channel is #eigen on irc.freenode.net (irc://irc.freenode.net).

Everybody's welcome to discuss Eigen-related topics or just chat. Some quick Eigen development chat happens over IRC, but the main place for Eigen development discussion remains the Eigen mailing list. Some user support also happens over IRC, but the main place for Eigen user support is the Users Forum (http://forum.kde.org/viewforum.php?f=74).

Contributing to Eigen

Eigen is written and maintained by volunteers. You can contribute in many ways to help: give support to new users, write and improve documentation, helping with bugs and other issues in bugzilla, discussing the design and the API, running tests and writing code. See our page on Contributing to Eigen for pointers to get you started.

Projects using Eigen

Feel free to add yourself! If you don't have access to the wiki or if you are not sure about the relevance of your project, ask at the #Mailing list.

Extensions, numerical computation

- Google's TensorFlow (http://tensorflow.org) is an Open Source Software Library for Machine Intelligence
- Google's Ceres (http://ceres-solver.org/) solver is a portable C++ library that allows for modeling and solving large complicated nonlinear least squares problems.
- The Manifold ToolKit MTK (http://openslam.org/MTK) provides easy mechanisms to enable arbitrary algorithms to operate on manifolds. It also provides a Sparse Least Squares Solver (SLoM) and an Unscented Kalman Filter (UKFoM).
- IFOPT (https://github.com/ethz-adrl/ifopt) is a modern, light-weight, Eigen-based C++ interface to Nonlinear Programming solvers, such as Ipopt and Snopt.
- CppNumericalSolvers (https://github.com/PatWie/CppNumericalSolvers) is a lightweight header-only library for non-linear optimization including various solvers: CG, L-BGFS-B, CMAes, Nelder-Mead.
- GTSAM (https://borg.cc.gatech.edu/projects/gtsam) is a library implementing smoothing and mapping (SAM) in robotics and vision, using factor graphs and Bayes networks.
- g2o (https://openslam.org/g2o.html) is an open-source C++ framework for optimizing graph-based nonlinear least-square problems.
- redsvd (http://code.google.com/p/redsvd/) is a RandomizED Singular Value Decomposition library for sparse or very large dense matrices.
- trustOptim (http://cran.r-project.org/web/packages/trustOptim) is a trust-region based non linear solver supporting sparse Hessians (C++ implementation with R binding).
- Shogun (http://shogun-toolbox.org/): a large scale machine learning toolbox.
- Stan (http://mc-stan.org/): a statistical package based on Eigen that includes a reverse-mode automatic differentiation implementation.
- StOpt (https://gitlab.com/stochastic-control/StOpt), the STochastic OPTimization library aims at providing tools in C++ for solving some stochastic optimization problems encountered in finance or in the industry.
- Ceemple (http://www.ceemple.com) is a scientific programming environment allowing to rapidly prototype applications C++ as in MatLab or Python.

- Nelson (https://nelson-numerical-software.github.io/nelson-website) an open computing environment for engineering and scientific applications using modern C/C++ libraries (Boost, Eigen, FFTW, …) and others state of art numerical libraries. (GPL2)
- EigenLab (https://github.com/marcel-goldschen-ohm/EigenLab) is a header only library to parse and evaluate expressions working on Eigen matrices.
- SpaFEDte (http://spafedte.github.io/) a C++ library for discontinuous Galerkin discretizations on general meshes.
- biicode (https://www.biicode.com/) a C and C++ dependency manager that #includes the most popular and useful C/C++ libs and frameworks.
- CERTH's AKSDA (http://mklab.iti.gr/project/aksda): a GPU-accelerated dimensionality reduction and classification method for high-dimensional data.
- Spectra (https://spectralib.org/) stands for Sparse Eigenvalue Computation Toolkit as a Redesigned ARPACK. It is a header-only C++ library for large scale eigenvalue problems, built on top of Eigen.
- preCICE (https://www.precice.org/) is a coupling library for partitioned multi-physics simulations, including, but not restricted to fluid-structure interaction and conjugate heat transfer simulations. It supports OpenFOAM, CalculiX, SU2, and several other well-known, as well as in-house solvers. It is free/open-source software and its code is available on GitHub under the LGPL3 license. Link to source code and documentation (https://github.com/precice/)

Bindings

- The RcppEigen (http://cran.r-project.org/web/packages/RcppEigen/index.html) package provides bindings and more for R.
- minieigen (https://pypi.python.org/pypi/minieigen) is small boost::python wrapper for Eigen's core classes (dense fixed and dynamic-sized vectors, matrices, aligned boxes, quaternions; with integer, floating and complex scalars) including operators and subset of Eigen's API methods.
- Eigency (https://pypi.python.org/pypi/eigency) is a Cython interface between the numpy arrays and the Matrix/Array classes of the Eigen C++ library.
- sparray (http://packages.python.org/sparray): a python binding of the Sparse module - alpha stage.
- OCamlEigen (https://github.com/ryanrhymes/eigen), a thin OCaml interface to the Eigen 3 C++ library.
- haskell-eigen (https://github.com/osidorkin/haskell-eigen) offers bindings for the Haskell language.
- jeigen (https://github.com/hughperkins/jeigen), a Java wrapper for Eigen.
- Eigen4AutoIt (https://www.autoitscript.com/forum/files/file/319-eigen4autoit/), a wrapper for the AutoIt scripting language.
- eigen-lua (https://github.com/ggcrunchy/eigen-lua), a Lua wrapper around parts of the Eigen numerical library.
- Eigenpy (https://github.com/stack-of-tasks/eigenpy): Efficient bindings between Numpy and Eigen using Boost.Python with support of the Geometry module

Science

- GINESTRA (http://www.mdlab-software.com/), a semiconductor device simulator with a focus on advanced dielectric materials and interfaces.
- G+Smo (http://gs.jku.at/gismo), an open-source library for geometric design and numerical simulation with isogeometric analyis.
- FlexibleSUSY (https://flexiblesusy.hepforge.org/), a spectrum generator which calculates the masses of elementary particles.
- The ATLAS (http://home.cern/about/experiments/atlas) experiment at the LHC (Large Hadron Collider) (http://home.cern/topics/large-hadron-collider) at CERN is using Eigen, as reported in this article (http://iopscience.iop.org/article/10.1088/1742-6596/608/1/012047/pdf), noting "Eigen was chosen since it offered the largest performance improvements for ATLAS use cases of the options investigated."
- The Large Survey Synoptic Telescope (website (http://www.lsst.org/lsst/); trac (https://dev.lsstcorp.org/trac)) is a project to build a 3.2Mpixel camera on an 8.4m telescope and survey the entire visible sky every three days.

- Gnu Data Language (http://gnudatalanguage.sourceforge.net/), a GPL interpretor of IDL syntax codes.
- The Space Trajectory Analysis (http://sta.estec.esa.int/Space_Trajectory_Analysis/Home.html) project at the European Space Agency (http://www.esa.int/esaCP/index.html). They even have an announcement on using Eigen (http://sta.estec.esa.int/Space_Trajectory_Analysis/News/Entries/2009/3/31_STA_astro-core_is_using_the_Eigen_library.html).
- Avogadro (http://avogadro.openmolecules.net), an opensource advanced molecular editor.
- The 3D astronomical visualization application Celestia (http://www.shatters.net/celestia) is now using Eigen for all orbital and geometric calculation.
- Yade (https://www.yade-dem.org/sphinx/), platform for dynamic particle models, uses Eigen for geometric computations (switched from the WildMagic (http://www.geometrictools.com) package)
- SLangTNG (https://tng.tuxfamily.org), an application suite for numerical analysis, linear algebra, advanced statistics, FEM, structural dynamics, data visualization, etc.
- Clip (http://clip.berlios.de), an opensource program for the orientation of Laue exposures.
- Multiprecision Computing Toolbox for MATLAB (http://www.advanpix.com/) uses Eigen as core library for matrix computations.
- Pteros (http://pteros.sourceforge.net/), a C++ library for molecular modeling.
- Cufflinks (http://cufflinks.cbcb.umd.edu/), a tool for transcript assembly, differential expression, and differential regulation for RNA-Sequences.
- Vortexje (http://www.baayen-heinz.com/vortexje): a GPL library for simulation implementing the source-doublet panel method.
- Woo(dem) (http://www.woodem.eu), particle dynamics software (DEM, FEM); Eigen wrapped using minieigen (https://pypi.python.org/pypi/minieigen) in Python.
- NIMBLE (http://R-nimble.org), a system for programming statistical algorithms such as Markov chain Monte Carlo from R. NIMBLE includes a compiler for a subset of R to C++ that uses Eigen.
- ENigMA (https://github.com/bjaraujo/ENigMA) is a multiphysics numerical library which uses Eigen.
- iMSTK (http://www.imstk.org/) is an open source software toolkit written in C++ that aids rapid prototyping of interactive multi-modal surgical simulations.
- mbsolve (https://github.com/mriesch-tum/mbsolve) is an open-source solver tool for the Maxwell-Bloch equations, which are used to model light-matter interaction in nonlinear optics.
- Spin-Scenario (https://github.com/spin-scenario/spin-scenario), a flexible scripting environment for realistic magnetic resonance (NMR/MRI) simulations.
- Quantum++ (https://github.com/vsoftco/qpp) is a modern C++11 general purpose quantum computing library, composed solely of template header files.

Robotics and engineering

- The Yujin Robot (http://yujinrobot.com/) company uses Eigen for the navigation and arm control of their next gen robots. (switched from blitz, ublas and tvmet)
- The Robotic Operating System (ROS) (http://www.ros.org) developed by Willow Garage (http://www.willowgarage.com).
- openAHRS (http://github.com/cbecker/openahrs) Open Source IMU (http://en.wikipedia.org/wiki/Inertial_Measurement_Unit) / AHRS (http://en.wikipedia.org/wiki/Attitude_and_Heading_Reference_Systems)
- The Darmstadt Dribblers (http://www.dribblers.de/) autonomous Humanoid Robot Soccer Team and Darmstadt Rescue Robot Team (http://www.gkmm.tu-darmstadt.de/rescue/) use Eigen for navigation and world modeling.
- The Mobile Robot Programming Toolkit (MRPT) (http://www.mrpt.org/), a set of libraries for SLAM, localization and computer vision, moved to Eigen (switched from home made math classes).
- RBDL (https://bitbucket.org/rbdl/rbdl/): a C++ library for rigid body dynamics.
- RL (http://roblib.sf.net) a self-contained C++ library for robot kinematics, motion planning and control.
- BTK (https://code.google.com/p/b-tk/) is a Biomechanical ToolKit, licensed under BSD whose primary goal is to propose a set of tools for the analysis of the human body motion which is independent of any acquisition system. It proposes bindings for Matlab/Octave and Python, and a GUI software called Mokka to visualize/analyze 3D/2D motion capture data.

- libpointmatcher (https://github.com/ethz-asl/libpointmatcher) is a "Iterative Closest Point" library for 3D mapping in robotics.
- RobOptim (http://www.roboptim.net/) is a modern, Open-Source, C++ library for numerical optimization applied to robotics.
- towr (https://github.com/ethz-adrl/towr) is a light-weight and extensible C++ library for trajectory optimization for legged robots.
- Pinocchio (https://github.com/stack-of-tasks/pinocchio): a fast and efficient Rigid Body Dynamics library
- The Humanoid Path Planner (https://github.com/humanoid-path-planner): a software for Motion and Manipulation Planning
- MIRA (http://www.mira-project.org) is a cross-platform framework written in C++ that provides a middleware for the development of complex robotic applications, which consists of distributed software modules.

Computer Graphics, Geometry, Visualization

- Computational Geometry Algorithms Library (CGAL) (http://www.cgal.org), a collaborative effort to develop a robust, easy to use, and efficient C++ software library of geometric data structures and algorithms.
- Point Cloud Library (PCL) (http://pointclouds.org), a large scale, BSD licensed, open project for point cloud processing. Uses Eigen as their math backend.
- VcgLib (http://vcg.sourceforge.net), an opensource C++ template library for the manipulation and processing of triangle and tetrahedral meshes. (switched from home made math classes)
- MeshLab (http://meshlab.sourceforge.net), an opensource software for the processing and editing of unstructured 3D triangular meshes and point cloud. (switched from vcglib's math classes)
- eos (https://github.com/patrikhuber/eos), an opensource and lightweight 3D Morphable Face Model fitting library in modern C++11/14.
- The Topology ToolKit (TTK) (https://topology-tool-kit.github.io/), an open-source library and software collection for topological data analysis in scientific visualization.
- Theia (http://www.theia-sfm.org/), an opensource C++ structure from motion library tailored for researchers, BSD licensed.
- libmv (http://code.google.com/p/libmv), an opensource structure from motion library. (switched from FLENS)
- openMVG (http://imagine.enpc.fr/~moulonp/openMVG), a simple library for multiple view geometry.
- metronome (https://github.com/ahupowerdns/metronome) , a simple metric storage/graphing engine using Eigen for SVD least squares interpolation & data consolidation
- piccante (http://piccantelib.net), a C++ imaging library designed for High Dynamic Range (HDR) processing.
- libigl (https://github.com/libigl/libigl/) is a simple C++ geometry processing library with wide functionality.
- ApproxMVBB (https://github.com/gabyx/ApproxMVBB) is a small library to compute fast approximate oriented bounding boxes of 3D point clouds.
- Madplotlib (https://github.com/madplotlib/madplotlib) makes it easier to plot 2D charts on Qt from data created by Eigen::ArrayXf.
- 3DF Zephyr (http://zephyr.3dflow.net) is a commercial photogrammetry and 3D modeling software, developed by 3Dflow srl (http://www.3dflow.net)

Mobile apps

- Layar (https://www.layar.com/), an augmented reality application for IPhone and Android.
- Red Sword Studios (http://www.redsword.com), maker of the iPhone games Gradient (http://linktoapp.com/Gradient), Fortress Luna (http://linktoapp.com/Fortress+Luna), Stimulus (http://linktoapp.com/Stimulus), and Lustre, uses Eigen extensively. Why roll your own matrix/vector/transformation code when there's Eigen?
- iPerfectPutt (http://www.iperfectputt.com/), an iPhone game, uses Eigen for 3D geometry calculations.
- WhirlyGlobe-Maply (http://mousebird.github.io/WhirlyGlobe) an open source geospatial display toolkit for iOS and Android. It implements both a 3D interactive globe and a 2D (slippy) map

KDE (our origins!)

- Step (http://edu.kde.org/step), an educational physics simulator.
- Koffice2 (http://www.koffice.org) (KDE's office suite), in particular Krita (http://www.koffice.org/krita/), the painting and image editing module. Eigen is also used a bit by KSpread (http://www.koffice.org/kspread/), the spreadsheet module, for matrix functions such as MINVERSE, MMULT, MDETERM.
- Kalzium (http://edu.kde.org/kalzium) uses Eigen indirectly through the aforementioned Avogadro (http://avogadro.openmolecules.net) library.
- the Mandelbrot wallpaper plugin, some screensavers, kgllib (http://freehackers.org/~rivo/kgllib/html/), solidkreator (http://websvn.kde.org/trunk/playground/graphics/solidkreator/), etc.

If you are aware of some interesting projects using Eigen, please send us a message (including a link and short description) or directly edit this wiki page!

Credits

The Eigen project was started by Benoît Jacob (founder) and Gaël Guennebaud (guru). Many other people have since contributed their talents to help make Eigen successful. Here's an alphabetical list: (note to contributors: **do add yourself!**)

Philip Avery	Fix bug and add functionality to AutoDiff module
Abraham Bachrach	Added functions for cwise min/max with a scalar
Sebastien Barthelemy	Fix EIGEN_INITIALIZE_MATRICES_BY_NAN
Carlos Becker	Wrote some of the pages of the tutorial
David Benjamin	Artwork: the owls
Cyrille Berger	Fix error in logic of installation script
Armin Berres	Lots of fixes (compilation warnings and errors)
Jose Luis Blanco	Build fixes for MSVC and AMD64, correction in docs
Mark Borgerding	FFT module
Romain Bossart	Updates to Sparse solvers
Kolja Brix	Added docs to Householder module, fixes for ARPACK wrapper, KroneckerProduct and GMRES
Gauthier Brun	Making a start with a divide-and-conquer SVD implementation
Philipp Büttgenbach	Bug fix when differentiating a one-dimensional spline
Thomas Capricelli (http://www.freehackers.org/thomas)	Migration to mercurial (http://www.freehackers.org/thomas/2009/05/18/feedback-about-converting-eigen2-to-mercurial/), Non-linear optimization and numerical differentiation, cron-job to update the online dox
Nicolas Carre	Making a start with a divide-and-conquer SVD implementation
Jean Ceccato	Making a start with a divide-and-conquer SVD implementation
Vladimir Chalupecky	CMake fix
Benjamin Chrétien	Bug fix for PolynomialSolver (linear polynomials), various typos
Andrew Coles	Fixes (including a compilation error)
Jeff "complexzeros"	Spline interpolation with derivatives
Marton Danoczy	MSVC compilation fix, support for ARM NEON with Clang 3.0 and LLVM-GCC
Jeff Dean	Fix in vectorized square root for small arguments

Georg Drenkhahn	CMake and STL fixes
Christian Ehrlicher	MSVC compilation fix
Martinho Fernandes	Make our aligned allocator support C++11
Daniel Gomez Ferro	Improvements in Sparse and in matrix product
Rohit Garg (http://rpg-314.blogspot.com/)	Vectorized quaternion and cross products, improved integer product
Mathieu Gautier	QuaternionMap and related improvements
Anton Gladky	Visual Studio 2008 and GCC 4.6 compilation fixes
Stuart Glaser	Prevent allocations in LU decomposition
Marc Glisse	C++11 compilation issues (suffices for literals)
Frederic Gosselin	Improve filter for hidden files in CMake
Christoph Grüninger	Add SuperLU 5 support
Gaël Guennebaud (http://www.labri.fr/perso/guenneba/)	Core developer
Philippe Hamelin	Allow CMake project to be included in another project
Marcus D. Hanwell	CMake improvements. Marcus is a developer at Kitware!
David Harmon	Arpack support module
Chen-Pang He	Many improvements to MatrixFunctions and KroneckerProduct modules
Hauke Heibel	Extended matrix functions, STL compatibility, Splines, CMake improvements, and more ...
Christoph Hertzberg (http://www.informatik.uni-bremen.de/agebv/en/ChristophHertzberg)	Quaternions, shifts for Cholmod, bug fixes, lots of user support on forums and IRC
Pavel Holoborodko (http://www.holoborodko.com/pavel/)	Multi-precision support with MPFR C++ (http://www.holoborodko.com/pavel/mpfr/), bug fixes for non-standard scalar types
Tim Holy	Improvements to tutorial, LDLT update and downdate
Intel	Back-end to Intel Math Kernel Library (MKL)
Trevor Irons	Square root for complex numbers, fix compile errors and mistake in docs
Benoît Jacob (http://people.mozilla.org/~bjacob)	Core developer
Bram de Jong	Improvement to benchmark suite
Kibeom Kim	Implement *= /= * / operations for VectorwiseOp
Moritz Klammler	Avoid memory leak when constructor of user-defined type throws exception
Claas Köhler	Improvements to Fortran and FFTW in CMake
Alexey Korepanov	Add RealQZ class
Igor Krivenko	Properly cast constants when using non-standard scalars
Marijn Kruisselbrink	CMake fixes
Abhijit Kundu	Fix compilation errors in OpenGL demos
Moritz Lenz (http://perlgeek.de/)	Allow solving transposed problem with SuperLU
Bo Li	Fix bugs in Spline constructor
Sebastian Lipponer	MSVC compilation support
Daniel Lowenberg	Add SparseView class

David J. Luitz	Bug fix for sparse * dense matrix product
Naumov Maks	Bug fix in polynomial solver test
Angelos Mantzaflaris	Fix to allow IncompleteLUT to be used with MPFR
D J Marcin	Fix operator& precedence bug
Konstantinos A. Margaritis	AltiVec/VSX, ARM NEON (32-bit/64-bit) and S390x SIMD (ZVector) vectorization
Roger Martin	Component-wise arc tangent function
Ricard Marxer	Reverse, redux improvements, the count() method, some dox
Vincenzo Di Massa	CMake fix
Christian Mayer	Early code review and input in technical/design discussions
Frank Meier-Dörnberg	MSVC compatibility fixes
Keir Mierle	LDLT decomposition and other improvements, help with MPL relicensing
Laurent Montel	CMake improvements. Laurent is (with Alexander) one of the CMake gurus at KDE!
Eamon Nerbonne	Compilation fixes for win32
Alexander Neundorf	CMake improvements. Alexander is (with Laurent) one of the CMake gurus at KDE!
Jason Newton	Componentwise tangent functions
Jitse Niesen (http://www.maths.leeds.ac.uk/~jitse/)	Matrix functions, large improvements in the Eigenvalues module and in the docs, and more ...
Desire Nuentsa (http://www.irisa.fr/sage/desire)	Many improvements to Sparse module: SparseLU, SparseQR, ILUT, PaStiXSupport, …
Jan Oberländer	Compatibility with termios.h
Jos van den Oever	Compilation fix
Michael Olbrich	Early patches, including the initial loop meta-unroller
Simon Pilgrim	Optimizations for NEON
Bjorn Piltz	Visual C compilation fix
Benjamin Piwowarski	Add conservativeResize() for sparse matrices
Zach Ploskey	Copy-editing of tutorial
Giacomo Po	MINRES iterative solver
Sergey Popov	Fix bug in SelfAdjointEigenSolver
Manoj Rajagopalan	Introduce middleRows() / middleCols(), bug fix for nonstandard numeric types
Stjepan Rajko	MSVC compatibility fix
Jure Repinc	CMake fixes
Kenneth Frank Riddile	Lots of Windows/MSVC compatibility fixes, handling of alignment issues
Richard Roberts	Bug fix in selection of MKL Householder QR
Adolfo Rodriguez	Prevent allocations in matrix decompositions
Peter Román	Support for SuperLU's ILU factorization
Oliver Ruepp	Bug fix in sparse matrix product with row-major matrices
Radu Bogdan Rusu	Fix compilation warning
Guillaume Saupin	Skyline matrices
Olivier Saut	Typo in documentation
Benjamin Schindler	gdb pretty printers

Michael Schmidt	Fix in assembly when identifying CPU, Compilation fix connected to min/max
Dennis Schridde	New typedefs like AlignedBox3f
Jakob Schwendner	Test for unaligned quaternions, Benchmark for Geometry module
Christian Seiler	The Tensor module
Martin Senst	Bug fix for empty matrices
Sameer Sheorey	Fix gdb pretty printer for variable-size matrices
Andy Somerville	Functions to get intersection between two ParametrizedLines
Alex Stapleton	Help with tough C++ questions
Benoit Steiner (http://bsteiner.info)	Faster matrix multiplication, Tensor module, CUDA
Sven Strothoff	Add intersects() method to AlignedBox
Leszek Swirski	Fix oversight in installation scripts
Adam Szalkowski	Bug fix in MatrixBase::makeHouseholder()
Silvio Traversaro	Fix for FindEigen3.cmake
Piotr Trojanek	QCC compilation fixes
Anthony Truchet	Bugfix in QTransform and QMatrix support
Adolfo Rodriguez Tsourouksdissian	Version of JacobiSVD that pre-allocates its resources
James Richard Tyrer	CMake fix
Rhys Ulerich	Pkg-config support, improved GDB pretty-printer
Henry de Valence	Typo in documentation
Ingmar Vanhassel	CMake fix
Michiel Van Dyck	Multi-threading documentation bug fix
Scott Wheeler	Documentation improvements
Freddie Witherden	CMake fixes, allow LevenbergMarquardt to work with non-standard types
Urs Wolfer	Fixed a serious warning
Manuel Yguel	Bug fixes, work on inverse-with-check, the Polynomial module
Pierre Zoppitelli	Making a start with a divide-and-conquer SVD implementation
Jonas Adler (https://www.kth.se/profile/jonasadl/)	Bugfixes for CUDA support

Eigen is also using code that we copied from other sources. They are acknowledged in our sources and in the Mercurial history, but let's also mention them here:

Intel Corporation	SSE code for 4x4 matrix inversion taken from here (http://software.intel.com/en-us/articles/optimized-matrix-library-for-use-with-the-intel-pentiumr-4-processors-sse2-instructions/).
Tim Davis (http://www.cise.ufl.edu/~davis)	AMD reordering simplicial sparse Cholesky factorization adapted from SuiteSparse (http://www.cise.ufl.edu/research/sparse/SuiteSparse)
Julien Pommier	SSE implementation of exp,log,cos,sin math functions from GMM++ (http://download.gna.org/getfem/html/homepage/gmm/index.html)
Yousef Saad (http://www-users.cs.umn.edu/~saad)	IncompleteLUT preconditioner coming from ITSOL (http://www-users.cs.umn.edu/~saad/software/ITSOL)
Minpack authors (http://en.wikipedia.org/wiki/MINPACK)	Algorithms for non linear optimization.

Special thanks to Tuxfamily for the wonderful quality of their services, and the GCC Compile Farm Project that gives us access to many various systems including ARM NEON.

If you are looking for a BibTeX entry to use to cite Eigen in academic papers, see the BibTeX page.

Retrieved from "http://eigen.tuxfamily.org/index.php?title=Main_Page&oldid=2432"

- This page was last modified on 15 July 2019, at 15:57.
- This page has been accessed 9,573,703 times.
-