

Computer program fixes old code faster than expert engineers

10 July 2015, by Adam Conner-Simons



The Helium program could help solve “a billion-dollar problem” in computing: companies having to devote time and energy to patching old code by hand. Credit: Christine Daniloff/MIT

Last year, MIT computer scientists and Adobe engineers came together to try to solve a major problem that many companies face: bit-rot.

A good example is Adobe's successful Photoshop photo editor, which just celebrated its 25th birthday. Over the years Photoshop had accumulated heaps of code that had been optimized for what is now old hardware.

"For high-[performance](#) code used for image-processing, you have to optimize the heck out of the software," says Saman Amarasinghe, a professor at MIT and researcher at the Computer Science and Artificial Intelligence Laboratory (CSAIL). "The downside is that the code becomes much less effective and much more difficult to

understand."

This results in what Amarasinghe describes as "a billion-dollar problem": companies like Adobe having to devote massive manpower to going back into the code every few years and, by hand, testing out a bunch of different strategies to try to patch it.

But what if there were a computer program that could automatically fix old code so that engineers can focus on more important tasks, such as actually dreaming up new software?

Enter Helium, a CSAIL system that revamps and fine-tunes code without ever needing the original source, in a matter of hours or even minutes.

The team started with a simple building block of programming that's nevertheless extremely difficult to analyze: binary code that has been stripped of debug symbols, which represents the only piece of code that is available for proprietary software such as Photoshop.

A particular type of computational kernel popular for such software are "stencil kernels," which allow you to do operations for entire areas of pixels. Stencil kernels are especially important to update because they use huge amounts of memory and compute power, and their performance degenerates quickly as new hardware become available.

With Helium, the researchers are able to lift these kernels from a stripped binary and restructure them as high-level representations that are readable in Halide, a CSAIL-designed [programming language](#) geared towards image-processing.

Going from binary to high-level languages was a big leap that the team originally didn't think was doable, according to lead author Charith Mendis.

"The order of operations in these optimized binaries are complicated, which means that they can be

hard to disentangle," says Mendis, a graduate student at CSAIL. "Because stencils do the same computation over and over again, we are able to accumulate enough data to recover the original algorithms."

From there, the Helium system then replaces the original bit-rotted components with the re-optimized ones. The net result: Helium can improve the performance of certain Photoshop filters by 75 percent, and the performance of less optimized programs such as Microsoft Windows' IrfanView by 400 to 500 percent.

"We've found that Helium can make updates in one day that would take human engineers upwards of three months," says Amarasinghe. "A system like this can help companies make sure that the next generation of code is faster, and save them the trouble of putting 100 people on these sorts of problems."

The research was presented in a paper accepted to the Association for Computing Machinery SIGPLAN conference on Programming Language Design and Implementation (PLDI 2015), which took place June 13-17 in Portland, Oregon.

The paper was written by Mendis, fellow graduate students Jeffrey Bosboom and Kevin Wu, research scientist Shoaib Kamil, postdoc Jonathan Ragan-Kelley PhD '14, Amarasinghe, and researchers from Adobe and Google.

"We are in an era where computer architectures are changing at a dramatic rate, which makes it important to write code that can work on multiple platforms," says Mary Hall, a professor at the University of Utah's School of Computing. "Helium is an interesting approach that has the potential to facilitate higher-level descriptions of stencil computations that could then be more easily ported to future architectures."

One unexpected byproduct of the work is that it lets researchers see the different tricks that programmers used on the old [code](#), such as archaeologists combing through computational fossils.

"We can see the 'bit hacks' that engineers use to optimize their algorithms," says Amarasinghe, "as well as better understand the larger context of how programmers approach different coding challenges."

More information: "Helium: Lifting High-Performance Stencil Kernels from Stripped x86 Binaries to Halide DSL Code."

[groups.csail.mit.edu/commit/pa ... is-pldi15-helium.pdf](https://groups.csail.mit.edu/commit/papers/pldi15-helium.pdf)

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology

APA citation: Computer program fixes old code faster than expert engineers (2015, July 10) retrieved 18 June 2019 from <https://phys.org/news/2015-07-code-faster-expert.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.