Fix Colormaps With RandR 1.2 Capable X.org Drivers

Name and Contact Information

Name: Isaac Kamga

Email: u2isaac@gmail.com

IRC Nick: Izakey, Izakee

LinkedIn,Sourceforge,Github Username: Isaac Kamga

Title

Fix Colormaps With RandR 1.2 Capable X.org Drivers

Synopsis

For a few decades now, games have been used to entertain humans and as a stress-alleviating tool in most homes. This project aims at repairing x11 colormaps in RandR 1.2 capable drivers such as Radeon drivers (HD 6850 and HD 8730) and Intel HD 4000, etc which are employed in gamma settings for three-dimensional games which use the Simple Directmedia Library (SDL). Without these fixes, gamma correction (which is used to vary the overall brightness of game imagery), will not work when games such as Urban Terror and Eurotruck Simulator are being played. Over the next 3 months, x11 colormaps will be repaired, tested on RandR 1.2 capable drivers, included into randr/*, hw/* as well as other portions of the X server's source code and documented appropriately.

Benefits To The Community

For over five (5) years, gamma corrections have not worked when games which use SDL are being played. The primary benefit of this project will be the effectiveness of gamma settings in games like Urban Terror, The Walking Dead and Eurotruck Simulator.

Deliverables

Pre Mid-term Period

- Repair and test colormaps with Radeon HD drivers viz HD 6850 and HD 8730 .

Post mid-term period

- Repair and test colormaps with Intel HD 4000 driver

- Document all repairs to RandR 1.2 capable drivers.
- Disseminate updates to RandR 1.2 by writing articles and conference papers which could be presented during the X Developers Conference.

Development Schedule

Below is a tentative timeline for the implementation of this project.

July 20th: Pre Community Bonding Period

(2 weeks)

- Installed Fedora 20 and updated necessary packages for development of X.
- Joined the xorg mailing list and #xorg-devel on IRC, introduced myself and asked for more information on fixing colormaps project.
- Developed EVoC project proposal

August 18th : Community Bonding Period

(1 week)

- Study the documentation of X server in general and RandR 1.2 in particular.
- Study and compile X server's source code.
- Discuss with developers and mentors on #xorg-devel to refine my project proposal.

August 25th to October 3rd : Pre Midterm Work Period

(2 Weeks)

- Continued investigation and fine-tuning of project proposal.
- Discussion with mentors and developers to fine-tune project proposal.
- Factoring the global gamma value/ramp exposed code in randr/* and hw/* to repair RandR 1.2 gamma corrections in Radeon drivers.
- Testing in-game gamma corrections using the XFree86-VidModeExtension on the Linux version of any game on this list.

(3 Weeks)

- Continue factoring the global gamma value/ramp code in randr/* and hw/* to repair RandR 1.2 gamma corrections in Radeon drivers.
- Testing in-game gamma corrections using the -gamma command line switch of the xgamma command on the Linux version of any game on this list.

(Mid-term Evaluation Week)

- Testing and verification of RandR 1.2 code.
- Documenting updates to colormaps and further testing.

- Filling necessary midterm evaluation forms

October 3rd to November 14th : Post Mid-term Work Period

(2 Weeks)

- Factoring code in randr/* and hw/* to repair RandR 1.2 gamma corrections in Intel driver.
- Testing in-game gamma settings using the XFree86-VidModeExtension and the -gamma command line switch of the xgamma command on the Linux version of any game on this list.

(**3 Weeks**)

- Further testing of the RandR 1.2 code using SDL-based games.
- Running my code through valgrind to make sure memory leaks are dealt with.
- Document all the updates to the X server's code base and its interactions with the aforementioned RandR 1.2 capable drivers.

(Final Evaluation Week)

- Disseminate updates to the X server code base by writing articles and conference papers which could be posted on websites like www.phoronix.com or even presented during Scientific gatherings like the X Developers Conference.
- Filling necessary final evaluation forms.

Project Description

RandR (which stands for resize and Rotate) is a communications protocol written as an extension to the X11 protocol which facilitate the ability to resize, rotate and reflect the root window of a screen. It enables Personal Computers and handheld devices to change their screen sizes to drive external monitors at different resolutions from their built-in screens. It's version 1.3 introduces new functionality such as dynamic hot-plugging for display devices and it is supported by some common free drivers like Nouveau, nv, Avivo and Radeon HD. RandR 1.2 requires support from the X server, the driver supporting the extension's specification and the xrandr library.

This project aims at repairing x11 colormaps in RandR 1.2 capable drivers such as Radeon drivers (HD 6850 and HD 8730) and Intel HD 4000, etc used in gamma settings of games which depend on the Simple directMedia library (SDL) like Urban Terror, Eurotruck Simulator,The Walking Dead just to name a few. Gamma settings (corrections) are used to control the overall brightness of images in the aforementioned games by varying the ratio of Red

to Green to Blue light.On the xs windows system, one can set gamma corrections by using the xgamma command with the -gamma option in the command-line interface.

In most computer devices, pixel color data is represented using the RGB color model in which red, Green and Blue light is mixed to produce a wider range of colors. Pixel color data is usually either stored directly in video memory or indirectly using a separate color Look-Up Table (LUT). The former approach (technique) of color management is called Direct Color while the latter technique is called Indexed Color. RandR 1.2 capable drivers which use the direct technique for color management work properly in games. For games which use the Simple DirectMedia library (SDL) however, gamma corrections do not work when the games are in play. A proposed solution would be the use of Indexed color management technique in which color date is first indexed in a colormap before it is loaded into video memory. The pixels of the game's images should rather contain an index in the colormap instead of color data itself. The per-CRTC Look-Up Table (LUT) is used to transform the pseudocolor numbers (indices) stored in each pixel of video memory into physical colors represented by RGB triplets. Color data may be stored using several color depths which determine the number of bits used to represent the entire color of a pixel or a single color component (Red, Green, Blue) of that pixel.

Although the most common color depths used today include 24-bit (True color) and 15/16-bit color (High color), we'll tailor our development to each of the following color depths as the implementation specifics deem fit ;

- 8-bit color
- 15/16-bit color (High Color)
- 18-bit color
- 24-bit color (True Color)
- 30/36/48-bit Color (Deep Color)

We also test our repairs of x11 colormaps on RandR 1.2 capable drivers such as;

- Radeon drivers viz HD 6850 and HD 8730
- Intel HD 4000

Related Work And References

1. Bug 27222 - Gamma settings not working in 3D games on X server, https://bugs.freedesktop.org/show_bug.cgi?id=27222
2. Wikipedia Entry on Indexed Color, http://en.wikipedia.org/wiki/Indexed_color
3. Wikipedia Entry on Look-Up Tables,http://en.wikipedia.org/wiki/Colour_look-up_table

4. Mailing list thread on Improving per-CRTC gamma support, http://cgit.freedesktop.org/xorg/xserver/commit/?id=91f73b79b7ae64e5b846d1efeb470bb61a913720

Biographical Information

I am a final year M.Sc. Computer Science student at the University Of Buea, holder of a Bachelor of Science degree in Mathematics, one of the first Google Summer of Code finalists from francophone Africa and Community Manager of Google Developers Group Buea. After working on various data structuring and algorithmic individual and team projects in my University, I implemented a heart primitive (basic shape) for the BRL-CAD project during the 2013 summer and was later on invited to Google's headquarters in Mountain View, CA as an Independent Consultant to co-author a Contributors guide to BRL-CAD. My capacity to excellently communicate in both English and French will enable me easily interact with mentors and help guide younger developers into the development of X windows system.

Projects

Google Summer Of Code Finalist, BRL-CAD, April to September 2013

- **I**mplemented a heart primitive shape and produced a cool animation. **C, imagemagick, over 1000 lines of code .**
- Tested the polynomial root solver for BRLCAD to ascertain its ability to solve sextic equations.
- Wrote new implementation of a Red black tree together with unit tests to handle light rays during ray tracing for BRLCAD following the 3rd Edition for Introduction to Algorithms by Cormen et al.

BookSprint Participant, Google Summer Of Code Doc Camp, Mountain View, CA, October 2013

- Coauthored Hacking BRL-CAD, a contributors guide to enable more developers, document writers and artists improve and contribute to the BRL-CAD software.

Legal Informatics (Individual Project),

- Built minicompilers to infer the compliance of business process graphs to government regulations. **C,C++, Flex/Bison. 500 lines of code**.

Skills

Languages: C (Excellent), **C++** (Excellent), **Bash** (Excellent), **Java** (Proficient).

Tools: Linux, Secure shell, Subversion,git, valgrind, emacs, gdb, flex/Bison.

Natural language : English (Excellent), French (Excellent).

Research

- A literature review of web mining, unpublished manuscript.
- A better algorithm for market basket analysis, unpublished manuscript.

Community Involvement

Manager, Google Developer Group Buea Cameroon.

- Organizing team, Google I/O Extended Buea 2013.
- Organized and Launched the Google Cloud Developer Challenge 2013 in Buea.
- Organizing team, Google Open Day at University Of Buea.
- Organizer, Buea Google Summer Of Code Meetup 2014.
- Organizer, Buea TechWomen Roundtable.
- Organizer, Google I/O Extended Buea 2014.

TIME AVAILABILITY

I'll be available to offer at least 40 hours a week for this project. I have finished writing my thesis and will be called up by my Department for my Pre-defense and Defense soon which should not take me more than 6 hours put together. I will endeavour to push code during night hours and weekends to meet up with project duties. I'll also regularly communicate the progress with my mentors by chatting on IRC, writing on the mailing list and updating my diary. I am also available to disseminate any research results which I shall gather during this project's implementation.