

# **CSIT6910A Independent Project**

**2012 Spring**

## **Firefox Browser Extension**

**Color Distance between Client and Server**

**Student: HONG Shuhua**

**E-mail: [shongaa@ust.hk](mailto:shongaa@ust.hk)**

**Supervisor: Prof. David Rossiter**

# Contents

<b>1. Introduction</b> .....	<b>1</b>
1.1 Overview.....	1
1.2 Objectives .....	1
<b>2. Preparation</b> .....	<b>2</b>
2.1 Firefox Browser Extension Structure.....	2
2.2 Used Languages .....	4
<b>3. Extension Design</b> .....	<b>5</b>
3.1 Basic Idea .....	5
3.2 Extension Three Main Parts .....	5
3.2.1 <i>IP Path Tracing</i> .....	5
3.2.2 <i>Create Picture</i> .....	6
3.2.3 <i>Dynamic Library Link Connection</i> .....	6
3.3 Extension Sub-functions.....	7
3.3.1 <i>Delete Pictures</i> .....	7
3.3.2 <i>Display IP Path Tracing Details</i> .....	8
<b>4. Implementation</b> .....	<b>9</b>
4.1 Generate Extension File and Installation .....	9
4.2 Operation .....	13
<b>5. Conclusion</b> .....	<b>14</b>
<b>6. Reference</b> .....	<b>15</b>
<b>7. Appendix</b> .....	<b>16</b>

# 1. Introduction

## 1.1 Overview

IP address or Domain Name is the identifier when user visits a website through the Internet. The user approaches to the target web page (on destination server) based on IP addresses of routers that link different networks. So, how far is between the user (host) and the target web pages (destination server). In order to know the distance between user and the location of the target web pages, one commonly use method is to tracing target web pages IP path.

Here, use the various color segments to represent the IP tracing path rather than just display the IP tracing path in number format, like 192.168.1.1.

## 1.2 Objectives

The idea of project is that to make a Firefox Browser extension, which uses various color's segments to express the route of web page's request and the width of each segments represent the time that the request on the specific part of the IP route. All color's segments together displays the distance between user and web page.

## 2 Preparation

### 2.1 Firefox Browser Extension Structure

The type of Firefox Browser Extension is .xpi file. It seems like a .zip file.

The .xpi file contain:

(1) Folders

- Components: the place for Dynamic Library Link file.
- Content: the place for the element files (.xul) of the extension and javascript files (.js) that controls main functions.

- Modules: the place for the .jsm files that provide interface and control to access to the Dynamic Library Link file and so on.
- Skin: the place for layout files (.css) and picture files (like, .jpeg)

## (2) Other files

- chrome.manifest: its content is to declare the above folders and files for Firefox Browser to recognize and overlay.

The content of chrome.manifest:

```
content colorWebDistance_Ext content/
overlay chrome://browser/content/browser.xul
chrome://colorWebDistance_Ext/content/overlay.xul
locale colorWebDistance_Ext locale/
skin colorWebDistance_Ext classic/1.0 skin/
style chrome://global/content/customizeToolbar.xul
chrome://colorWebDistance_Ext/skin/sample.css
component colorWebDistance_Ext components/
resource colorWebDistance_Ext modules/
```

- install.rdf: it is used as identifier and metadata of the extension of Firefox Browser

The content of install.rdf:

```
<?xml version="1.0"?>
<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:em="http://www.mozilla.org/2004/em-rdf#">
<Description about="urn:mozilla:install-manifest">
<em:id>sofa_hong103@colorDistance.com</em:id>
```

```
<em:unpack>true</em:unpack>

<em:version>1.0</em:version>

<em:type>2</em:type>

<!-- Target Application this extension can install into,
      with minimum and maximum supported versions. -->

<em:targetApplication>

  <Description>

<em:id>{ec8030f7-c20a-464f-9b0e-13a3a9e97384}</em:id>

  <em:minVersion>1.5</em:minVersion>

  <em:maxVersion>11.0.*</em:maxVersion>

  </Description>

</em:targetApplication>

<!-- Front End MetaData -->

<em:name>colorWebDistance_Ext</em:name>

<em:description>Use the color changes to express the distance
between your computer and the web pages</em:description>

<em:creator>HSH</em:creator>

<em:homepageURL>http://www.baidu.com/</em:homepageURL>

  </Description>

</RDF>
```

## 2.2 Used Languages

### (1) C++

C++ is one of the most popular programming languages with application domains including systems software, application software and so on.

Here, use it to build the DLL that executes the main functions, like IP route trace.

### (2) XML User Interface Language (XUL)

It is an XML user interface markup language developed by the Mozilla. XUL operates in Mozilla cross-platform applications such as Firefox.

Here, use it to create the control elements, like, button and drop down list.

### (3) Cascading Style Sheet (CSS)

It is a style sheet language used for describing the appearance of a document written in a markup language.

Here, use it to control the appearance of the control elements of the extension, like the width and so on.

### (4) Javascript

It is a prototype-based scripting language that is dynamic, weakly typed and has first-class functions.

Here, use it to connect the DLL and create sub functions for better performance in the Firefox Browser.

## **3 Extension Design**

### **3.1 Basic Idea**

Each passed routers is represented by one color using RGB and then the user side presents all color segments between routers according to the interval time of each router's response and displays them as a small colorful rectangle picture.

So, create a Dynamic Library Link (DLL) to operate the IP route from user to target web page and create a small picture of the all color segments

Then user is available to control how these color segments is displayed through choosing the creating pattern. (Like, use the IP first part to represent the color channel "R".) Finally, put the extension at the bottom of Firefox Browser as part of its components.

### **3.2 Extension Three Main Parts**

#### **3.2.1 IP Path Tracing**

Use C++ programming language to simulate the web page request to the server that target web page.

First, user side will create IP packets. Then use the Windows sockets to send out packets to passed routers and turn on the timer. Then, wait the passed router responding. When user side receives the reply packet, extract IP address of the router and stop the timer. Here, get two piece of information: passed router IP addresses and its response time.

### 3.2.2 Create Picture

With the passed router IP addresses, divide each IP address into four parts by one byte (like, 192.168.1.1). According to the create picture pattern user selects, use three of four parts to represent the color channel RGB. (Example, byte1: "192" represent channel "R"; byte2: "168" represent channel "G"; byte3: "1" represent channel "B")

With the help of OpenCV (Open Source Computer Vision Library, it is a library of programming functions mainly aimed at real time computer vision), create the color segments according to the response time information and then combine them into a picture file (.jpeg).

### 3.2.3 Dynamic Library Link Connection

Firefox provides "AddonManager.jsm" and "ctypes.jsm" (".jsm" is a Javascript file that offer a serial functions) to better perform in the extension using the Javascript. "AddonManager.jsm" provides function to access the extension directory after the extension install in the browser. "ctypes.jsm" provide various functions to connect the DLLs.

With the help of "ctypes.jsm", after collecting the control information (like, create pattern), call the DLL function to create the corresponding picture. Then, with the help of "AddonManager.jsm", get the extension install directory and then store the picture from DLL in that directory and display it on the Firefox Browser.

Some pieces of code in Javascript:



```
/*Begin*/  
  
Components.utils.import("resource://gre/modules/ctypes.jsm");  
  
Components.utils.import("resource://gre/modules/AddonManager.jsm");  
AddonManager.getAddonByID("sofa_hong103@colorDistance.com",  
function(addon){  
  
var  
  
images_Path=addon.getResourceURI("skin/images/").QueryInterface(Components.interfaces.nsIFileURL).file.path;  
  
... ..  
  
var  
  
dll=ctypes.open(addon.getResourceURI("components/colorWebDistance.dll").QueryInterface(Components.interfaces.nsIFileURL).file.path);  
  
var create_png =  
  
dll.declare("createPNG",ctypes.winapi_abi,ctypes.char.ptr,ctypes.char.ptr,ctypes.int32_t,ctypes.char.ptr,ctypes.int32_t,ctypes.int32_t,ctypes.int32_t);  
  
var  
  
result=create_png(images_Path2,win_width,domain,tabIndex,rgb,step);  
  
var txt_name_func = dll.declare("return_txt_name",ctypes.winapi_abi,ctypes.char.ptr);  
  
/*End*/
```

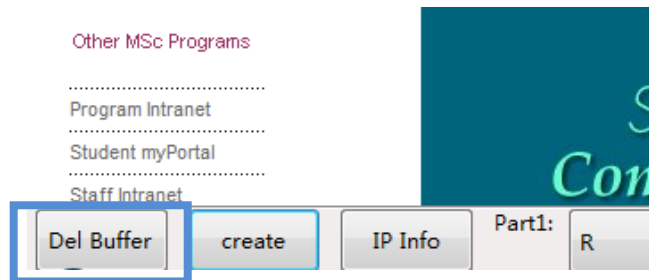
### 3.3 Extension Sub-functions

#### 3.3.1 Delete Pictures

Since the main function that creating color segments' picture will create a picture file store in the extension install directory. In order to make this

extension user-friendly, here create this function to delete the pictures.

This function mainly uses the C++ language and embeds in the DLL. Then it uses Javascript to connect the DLL, so that it can execute in the extension.



Delete Button

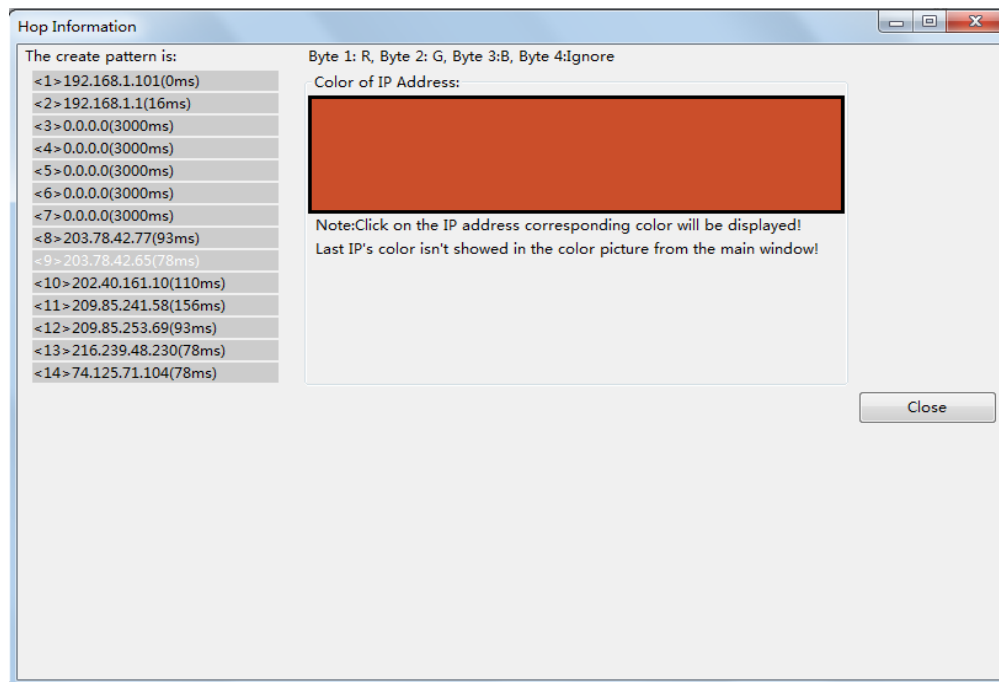


Delete Succeeded

### 3.3.2 Display IP Path Tracing Details

In order to let the user know more about the IP route color information, create this function to see host, destination server and each passed routers IP addresses and the color they represents. This function uses Javascript to change the background color of XUL tag- <box>, when user

click on the IP address.



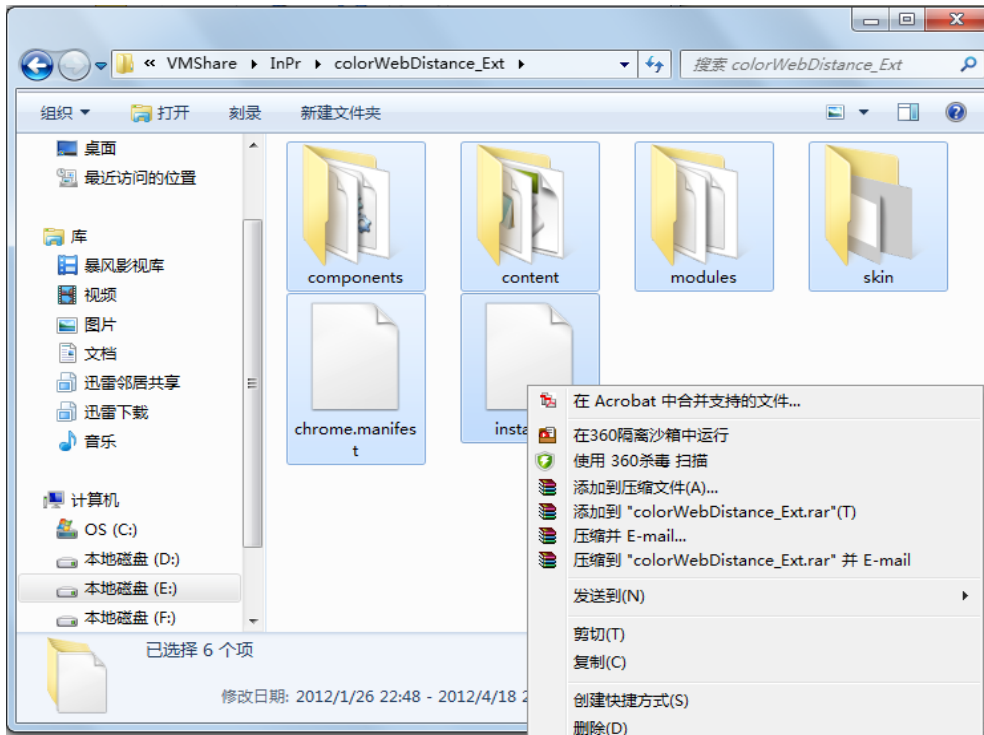
The Pop up Window

## 4 Implementation

### 4.1 Generate Extension File and Installation

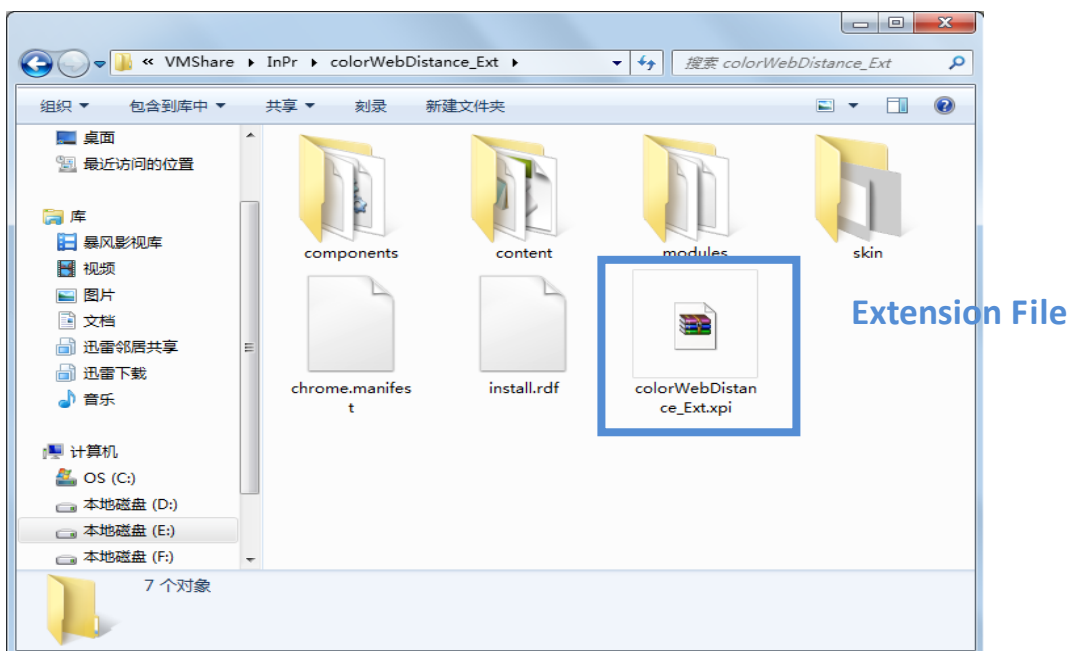
- Generate Extension File

With the structure folders and files show in the below picture, select all of them and right click the mouse. Then select to compress them as a .zip file.



Select Files

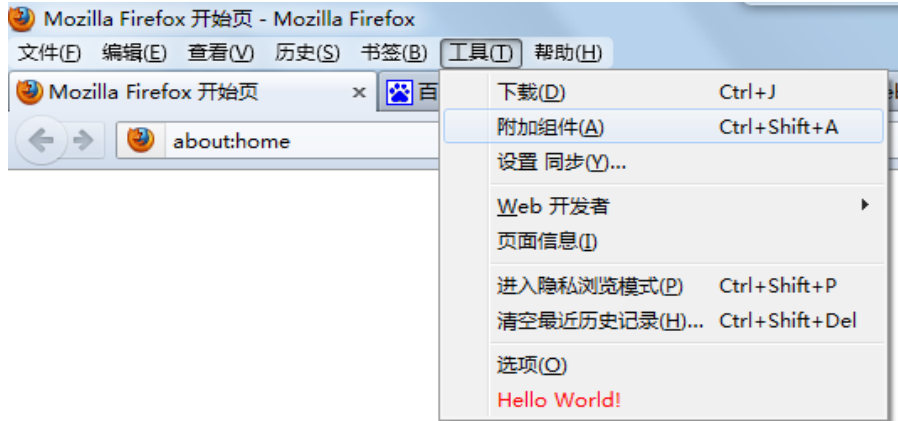
After generate the .zip file, change its name to .xpi file, which is the Firefox Browser extension file.



Extension File

- Installation

Install the extension file into the Firefox Browser. Select the “Tool” on the menu and get into the “add-ons” window.



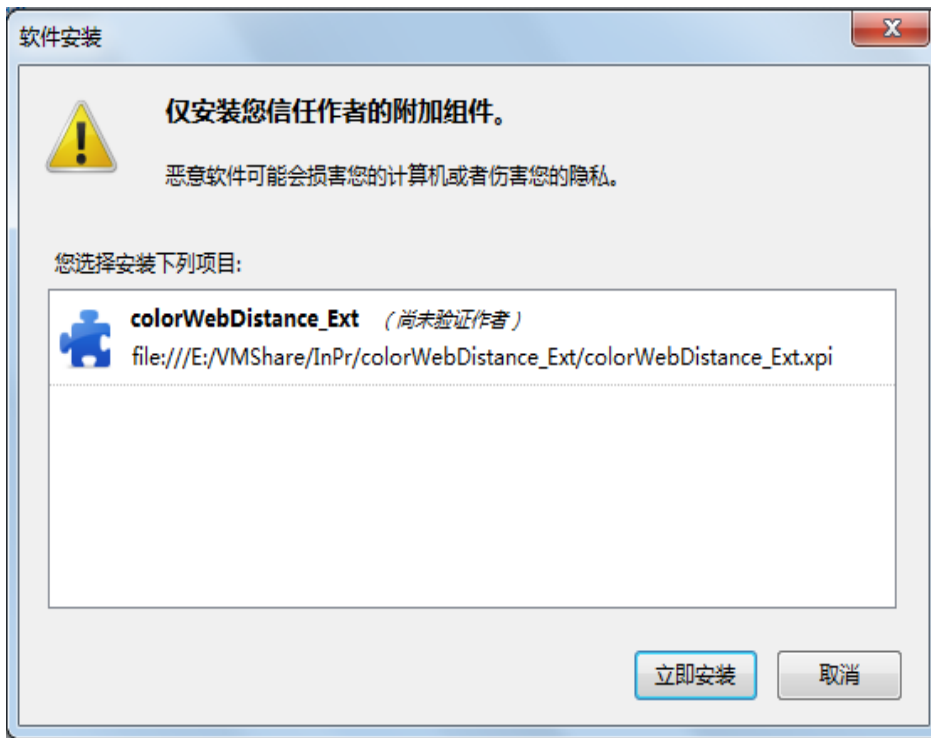
Install 01

After get into the “add-ons” window, select the extension file into the browser.



Install 02

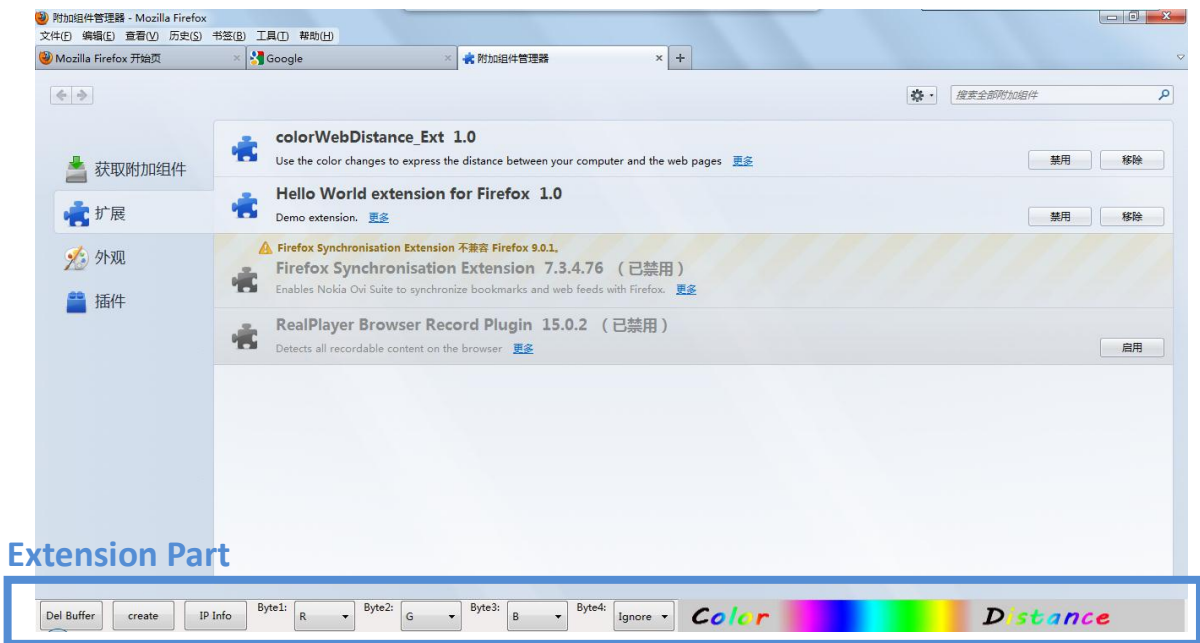
The browser will require user to admit the installation.



Install 03

After installation, Firefox Browser requires restarting the browser.

After restart the Firefox Browser, it will display like below:



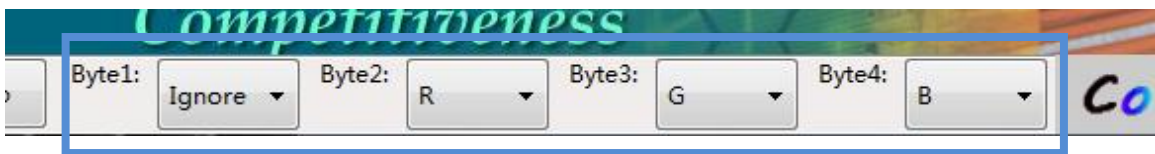
Extension displayed After Installation

## 4.2 Operation

Notice: Before operate this extension, user should run the Firefox Browser as administrator and close the window firewall if user is using Windows7. Otherwise it will not create the color distance picture.

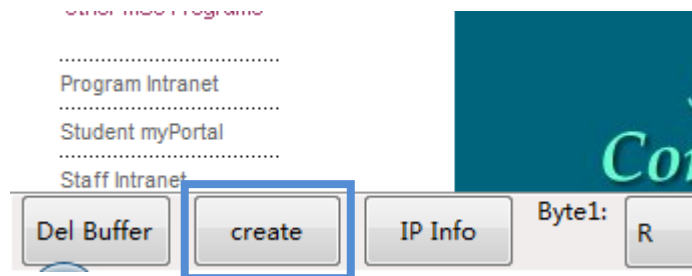
Select the create pattern first or let it be default pattern.

(Default Pattern: Byte1: R, Byte2: G, Byte3:B, Byte4:Ignore)



Select Create Pattern

Click the “create” button:



Create Color Distance

The created color distance picture is displayed below.



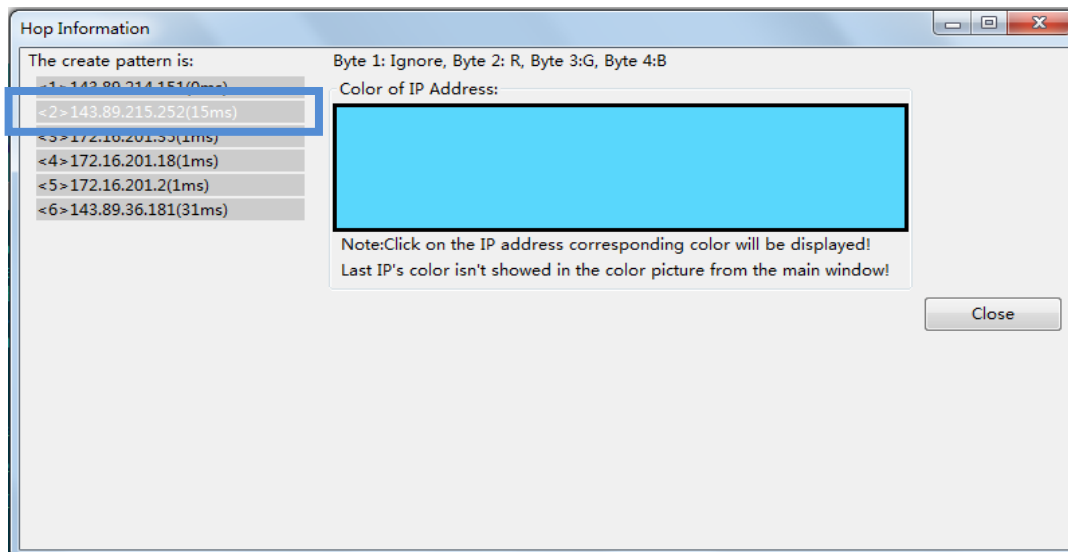
Created Color Distance

The width of the each color segments in the color distance picture represents the percentage of the time that one router (or host) hops to next router (or server) over the time that the host information goes to server in total.

Then user can see IPs trace and the detail information of each color segment from the color distance picture just created.



The Button to See Details



Details Window

In the above case, since using the create pattern-Byte1: Ignore; Byte2: R; Byte3: G; Byte4: B, the color displayed in the box is RGB(89,215,252). The number in brackets is the time one router (or host) hops to next router (or server) spends.

## 5 Conclusion

This Firefox Browser extension of color distance is aimed to display the distance between the user and the target page as color picture, which have various color segments. It bases on the IP router trace. Use parts of IP address to represent the color of a passed router and display the color



picture in the Firefox Browser as a part of component. With the aid of C++ and OpenCV this extension is accomplished successfully.

During the process of project, I tackled some challenges, such as building the Dynamic Link Library. Although there are several technologies I have not accessed before, I have learned them and used them in the project, like the language XUL.

However, this extension still needs future work to make it operate on cross platforms, like on Mac or Unix, so that more users can enjoy this extension.

## 6 Reference

- [1] <https://developer.mozilla.org/en>
- [2] <http://code.opencv.org/projects/OpenCV/wiki/WikiStart>
- [3] <http://www.rietta.com/firefox/Tutorial/guis.html>
- [4] <https://wiki.mozilla.org/JSctypes>
- [5] *How to do everything with JavaScript* Scott Duffy McGraw-Hill/Osborne  
2003
- [6] Introduction to programming with OpenCV.  
<http://www.cs.iit.edu/~agam/cs512/lect-notes/opencv-intro/opencv-intro.html#SECTION00050000000000000000>. Retrieved on 10 December 2011
- [7] *C++, the core language* Gregory Satir and Doug Brown 1st ed. O'Reilly & Assoc., Inc., 1995

## 7 Appendix

### Minutes of the 1<sup>st</sup> Project Meeting

**Date:** 20 Feb. 2012 (Monday)

**Time:** 9:45 am

**Place:** Room 3512

**Attending:** Prof. David Rossiter, HONG Shuhua

**Absent:** None

**Recorder:** HONG Shuhua

#### 1. Approval of Minutes

Since this is the first meeting, there is no approval of minutes of previous meeting.

#### 2. Report on Progress

Make a console application to simulate the command-trace in Windows.

#### 3. Discussion Items and Things To Do

- Project basic idea and other functions
- Ways of displaying the color

#### 4. Meeting Adjournment

The meeting was adjourned at 10:00 a.m.

## Minutes of the 2<sup>nd</sup> Project Meeting

**Date:** 12 Mar. 2012 (Monday)

**Time:** 9:45 am

**Place:** Room 3512

**Attending:** Prof. David Rossiter, HONG Shuhua

**Absent:** None

**Recorder:** HONG Shuhua

### 1. Approval of Minutes

The minutes of the last meeting were approved without amendment.

### 2. Report on Progress

Make the Dynamic Link Library about the function: to create color transformation picture and to display the trace route information.

### 3. Discussion Items and Things To Do

- To optimize the color transformation picture
- To increase the speed of the trace route function
- Preparation of making Firefox Browser extension

### 4. Meeting Adjournment

The meeting was adjourned at 10:05 a.m.

## Minutes of the 3<sup>rd</sup> Project Meeting

**Date:** 26 Mar. 2012 (Monday)

**Time:** 9:45 am

**Place:** Room 3512

**Attending:** Prof. David Rossiter, HONG Shuhua

**Absent:** None

**Recorder:** HONG Shuhua

### 5. Approval of Minutes

The minutes of the last meeting were approved without amendment.

### 6. Report on Progress

Finished the Firefox browser extension .xpi file basically.

### 7. Discussion Items and Things To Do

- To optimize the display of extension in Firefox browser
- Introduction of other way to speed up the extension process
- Preparation of independent project report and video

### 8. Meeting Adjournment

The meeting was adjourned at 10:10 a.m.

## Minutes of the 4<sup>th</sup> Project Meeting

**Date:** 23 Apr. 2012 (Monday)

**Time:** 10:00 am

**Place:** Room 3512

**Attending:** Prof. David Rossiter, HONG Shuhua

**Absent:** None

**Recorder:** HONG Shuhua

### 9. Approval of Minutes

The minutes of the last meeting were approved without amendment.

### 10. Report on Progress

The Firefox Browser extension was finished. The report of the project begins. The first version of video presentation is completed.

### 11. Discussion Items and Things To Do

- To optimize extension of the layout
- To optimize the presentation video
- To finish the report

### 12. Meeting Adjournment

The meeting was adjourned at 10:25 a.m.