

# λvue Sample GUI User Manual

v1.0.8

# Table of Contents

1. Introduction.....	1
1.1. What is λvue .....	1
1.2. About this guide .....	1
2. Linux.....	2
2.1. System Requirements .....	2
2.2. Qt Installation .....	2
2.3. Third-party Libraries Installation .....	2
2.4. Build λvue Sample GUI Application .....	3
3. Windows.....	4
3.1. System Requirements .....	4
3.2. Qt Installation .....	4
3.3. Build λvue Sample GUI Application .....	4
4. Mac .....	6
4.1. System Requirements .....	6
4.2. Qt Installation .....	6
4.3. Build λvue Sample GUI Application .....	6

The λvue Sample GUI User Manual provides an easy way to build a GUI application to use λvue SDK on Linux, Mac and Windows.



The λvue Sample GUI User Manual is still under construction, so the content of this document may change over time.

# 1. Introduction

This part gives you an overview of the technologies described in this guide.

## 1.1. What is λ·vue

Quanta Research Institute and MIT researchers collaborated to develop λvue that amplifies color changes and movements in video that are invisible to the naked eye. The λvue allows you to 'see' a person's pulse by magnifying the color of the person's skin, and to 'see' a person breathing by amplifying the subtle changes in movement.

Magnification enhances small movements in the source video. It works best for periodic changes such as breathing or pulse. This technique is designed to expose very small changes when the rest of the image is very still.

Quanta Research Institute has created this software development kit so that you can experiment with the λvue on your own videos and webcam.

For more information on the technology, please contact QRI ([info@lambda.qrilab.com](mailto:info@lambda.qrilab.com)).

## 1.2. About this guide

This guide covers the basic instructions needed to build a GUI application to use λvue SDK.

These instructions are intended to be used on λvue SDK installed machine.

# 2. Linux

## 2.1. System Requirements

The λvue sample GUI relies on tight integration with the host development environment, including the host compiler and C runtime libraries, and is therefore only supported on distribution versions that have been qualified for this λvue SDK.

### 2.1.1. Minimum System Requirements

- 2 GHz or faster 64-bit (x64) processor
- 4 GB RAM
- 2 GB available hard disk space



Laptop which has both Intel and NVIDIA graphics cards might have some issues of extension **"GLX" missing on display**

### 2.1.2. Environment Tested

- Ubuntu 12.04 64-bit
- Qt 5.5.0 for Linux 64-bit

## 2.2. Qt Installation

### 2.2.1. Ubuntu

We recommend to use the [Qt 5.5.0 for Linux 64-bit](#).

1. Download the [Qt 5.5.0 for Linux 64-bit installer](#).
2. Change mode of the installer and launch it with the steps below:

```
chmod +x qt-opensource-linux-x64-5.5.0-2.run
sudo ./qt-opensource-linux-x64-5.5.0-2.run
```

3. Follow the instructions to finish installation

## 2.3. Third-praty Libraries Installation

### 2.3.1. Ubuntu

1. Please follow the steps below to install third-party libraries:

```
sudo apt-get install libglu1-mesa-dev libconfig++8-dev
```

## 2.4. Build λvue Sample GUI Application

### 2.4.1. Ubuntu

1. Download **GUI Sample (All Platform)** from the [λvue webiste](#)
2. Decompress the tarball
3. Launch **Qt Creator** from the **Dash Home**
4. In the **Welcome** tab, click the **Open Project** button
5. Open the **GekoQt.pro** which located in the **LambdaVueGUI** directory
6. Click the **Configure Project** button
7. Select **Build** from the toolbar and click the **Build All**

The builds should produce no error message. The resulting binary will appear under the same directory of **GekoQt.pro** called **LambdaVueGUI**.

Once you compile finished, you can download our sample video from following URL and use this video to test λvue sample GUI by double click the resulting binary **LambdaVueGUI** or by typing following at the command line :

```
./LambdaVueGUI
```

- sample video : <http://lambda.qrilab.com/uploads/tutorial.mp4>

# 3. Windows

## 3.1. System Requirements

The  $\lambda$ vue sample GUI relies on tight integration with the host development environment, including the host compiler and C runtime libraries, and is therefore only supported on distribution versions that have been qualified for this  $\lambda$ vue SDK.

### 3.1.1. Minimum System Requirements

- 2 GHz or faster 64-bit (x64) processor
- 4 GB RAM
- 2 GB available hard disk space

### 3.1.2. Environment Tested

- Windows 7 32-bit / 64-bit
- Qt 5.4.2/5.2.1 with MinGW and OpenGL



In Windows system, you need to build with Qt, MinGW and OpenGL and it can run on 64-bit environment.

## 3.2. Qt Installation

We recommend to use the [Qt 5.4.2 with MinGW and OpenGL](#) or [Qt 5.2.1 with MinGW and OpenGL](#).

1. Download the [Qt 5.4.2 with MinGW and OpenGL](#) or [Qt 5.2.1 with MinGW and OpenGL](#).
2. Double click on the downloaded file
3. Follow the instructions to finish installation

## 3.3. Build $\lambda$ vue Sample GUI Application

1. Download **GUI Sample (All Platform)** from the  [\$\lambda\$ vue webiste](#)
2. Decompress the tarball
3. Copy dll files listed below from Qt/MinGW installation path to the LambdaVueGUI directory:



The dll files would located in a path like C:\Qt\Qt5.4.2\5.4.2\mingw49\_32\bin

- icuin\*.dll

- icuuc\*.dll
- icudt\*.dll
- libgcc\_s\_dw2-1.dll
- libstdc++-6.dll
- libwinpthread-1.dll
- Qt5Core.dll
- Qt5Gui.dll
- Qt5OpenGL.dll
- Qt5Widgets.dll

4. Launch **Qt Creator** from the **Start** menu
5. In the **Welcome** tab, click the **Open Project** button
6. Open the **GekoQt.pro** which located in the **LambdaVueGUI** directory
7. Click the **Configure Project** button



The default installation path of LambdaVue SDK is **C:\LambdaVueSDK**. If you have changed the installation path, please remember to modify your `SDK_ROOT` in line 23 of your `Geko.pro`.

8. Select **Build** from the toolbar and click the **Build All**

The builds should produce no error message. The resulting binaries will appear under the same directory of **GekoQt.pro** called `LambdaVueGUI.exe`.

Once you compile finished, you can download our sample video from following URL and use this video to test `λvue` sample GUI by double click on the **LambdaVueGUI.exe**.

- sample video : <http://lambda.qrilab.com/uploads/tutorial.mp4>

# 4. Mac

## 4.1. System Requirements

The λvue sample GUI relies on tight integration with the host development environment, including the host compiler and C runtime libraries, and is therefore only supported on distribution versions that have been qualified for this λvue SDK.

### 4.1.1. Minimum System Requirements

- 2 GHz or faster Intel-based processor
- 4 GB RAM
- 2 GB available hard disk space

### 4.1.2. Environment Tested

- Mac OS X 10.9 or later
- Xcode



Before you install Qt, you need install Xcode first. You can download Xcode from App Store. But it may require you to use the newest version of Mac OS X.

- Qt 5.5.0 for Mac

## 4.2. Qt Installation

We recommend to use the [Qt 5.5.0 for Mac](#).

1. Download the [Qt 5.5.0 for Mac installer](#).
2. Follow the instructions to finish installation

## 4.3. Build λvue Sample GUI Application

1. Download **GUI Sample (All Platform)** from the [λvue webiste](#)
2. Decompress the tarball
3. Launch **Qt Creator** from directory where you installed the Qt.
4. In the **Welcome** tab, click the **Open Project** button
5. Open the **GekoQt.pro** which located in the **LambdaVueGUI** directory
6. Click the **Configure Project** button



7. Select **Build** from the toolbar and click the **Build All**

The builds should produce no error message. The resulting binaries will appear under the same directory of **GekoQt.pro** called LambdaVueGUI.

Double click the LambdaVueGUI to open the sample application.



Once you compile done, you can download our sample video from following URL and use this video to test λvue sample GUI

- sample video : <http://lambda.qrilab.com/uploads/tutorial.mp4>