



Royale 3.21.1.70

Introduction

Version 3.21.1.70



Technical information subject to change without notice.

This document may also be changed without notice.

3.21.1.70

Created: 19.12.2018

©**pmd**technologies gmbh

All texts, pictures and other contents published in this instruction manual are subject to the copyright of **pmd**, Siegen unless otherwise noticed. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the publisher **pmd**.



Contents

1	Introduction	4
1.1	Operating Systems	4
1.2	Hardware Requirements	4
2	Getting Started	5
3	SDK and Examples	5
3.1	Debugging in Microsoft Visual Studio	5
4	Matlab	5
5	Reference	5
6	License	5

1 Introduction

The **Royale** software package provides a light-weight camera framework for time-of-flight (ToF) cameras. While being tailored to PMD cameras, the framework enables partners and customers to evaluate and/or integrate 3D TOF technology on/in their target platform. This reduces time to first demo and time to market.

Royale contains all the logic which is required to operate a ToF based camera. The user need not care about setting registers, but can conveniently control the camera via a high-level interface. The Royale framework is completely designed in C++ using the C++11 standard.

Royale officially supports the **CamBoard pico flexx**, **CamBoard pico maxx** and **CamBoard pico monstar** cameras.

1.1 Operating Systems

Royale supports the following operating systems:

- Windows 7/8
- Windows 10
- Linux (tested on Ubuntu 16.04)
- OS X (tested on El Capitan 10.11)
- Android (minimum android version 4.1.0. Tested on Android 5.1.1 and 7)
- Linux ARM (32Bit version tested on Raspbian GNU/Linux 8 (jessie) Raspberry Pi reference 2016-03-18 64Bit version tested on the Odroid C2 with Ubuntu Mate 16.04 ARM 64)

1.2 Hardware Requirements

Royale is tested on the following hardware configurations:

- PC, Intel i5-660, 3.3 GHz, 2 cores (64 bit)
- PC, Intel i7-2600, 3.4 GHz, 4 cores (64 bit)
- PC, Intel i7-3770, 3.4 GHz, 4 cores (64 bit)
- MacBook Pro Late 2013, Intel Core i5, 2.4 Ghz (64 bit)
- Samsung Galaxy S7
- Samsung Galaxy S8
- OnePlus 2
- Nexus 6
- Huawei Mate 9
- Raspberry Pi 3
- Odroid C2



2 Getting Started

For a detailed guide on how to get started with Royale and your camera please have a look at the corresponding Getting Started Guide that can be found in the top folder of the package you received.

3 SDK and Examples

Besides the royaleviewer application, the package also provides a Royale SDK which can be used to develop applications using the PMD camera.

There are multiple samples included in the delivery package which should give you a brief overview about the exposed API. You can find an overview in `samples/README_samples.md`. The `doc` directory offers a detailed description of the Royale API which is a good starting point as well. You can also find the API documentation by opening the `API_Documentation.html` in the topmost folder of your platform package.

3.1 Debugging in Microsoft Visual Studio

To help debugging `royale::Vector`, `royale::Pair` and `royale::String` we provide a Natvis file for Visual Studio. Please take a look at the `natvis.md` file in the `doc/natvis` folder of your installation.

4 Matlab

In the delivery package for Windows you will find a Matlab wrapper for the Royale library. After the installation it can be found in the `matlab` subfolder of your installation directory. To use the wrapper you have to include this folder into your Matlab search paths. We also included some examples to show the usage of the wrapper. They can also be found in the `matlab` folder of your installation.

5 Reference

FAQ: <http://pmdtec.com/picofamily/>

6 License

See `royale_license.txt`.

Parts of the software covered by this License Agreement (`royale_license.txt`) are using `libusb` under LGPL 2.1, `QT5.5` under LGPL 3.0, `gradle wrapper` under the Apache 2.0 license and `CyAPI` under the Cypress Software License Agreement (`cypress_license.txt`). `Spectre` is using `Kiss FFT` licensed under a BSD-style license and `PackedArray` licensed under the WTFPLv2. The documentation is created using `Doxygen`, which uses `jquery` and `sizzle`, both are licensed under the MIT license. The text of the GPL 3.0 is also provided because the LGPL 3.0, although more permissive, is written as a supplement to the GPL 3.0 and requires both texts.

The v3.11.x long-term support branch also contains `libuvc` under a BSD-style license.

Royale 3.21.1.70

Introduction



The source code of the open source software used in Royale is available at <https://oss.pmdtec.com/>.