RKNN Demo Developer Guide

Document ID: RK-KF-YF-334

Release Version: V1.0.0

Release Date: 2020-03-12

E-mail: wxt@rock-chips.com

DISCLAIMER

THIS DOCUMENT IS PROVIDED “AS IS”. FUZHOU ROCKCHIP ELECTRONICS CO., LTD. (“ROCKCHIP”)DOES NOT PROVIDE ANY WARRANTY OF ANY KIND, EXPRESSED, IMPLIED OR OTHERWISE, WITH RESPECT TO THE ACCURACY, RELIABILITY, COMPLETENESS,MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR NON-INFRINGEMENT OF ANY REPRESENTATION, INFORMATION AND CONTENT IN THIS DOCUMENT. THIS DOCUMENT IS FOR REFERENCE ONLY. THIS DOCUMENT MAY BE UPDATED OR CHANGED WITHOUT ANY NOTICE AT ANY TIME DUE TO THE UPGRADES OF THE PRODUCT OR ANY OTHER REASONS.

Trademark Statement

"Rockchip", "瑞芯微", "瑞芯" shall be Rockchip’s registered trademarks and owned by Rockchip. All the other trademarks or registered trademarks mentioned in this document shall be owned by their respective owners.

Fuzhou Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian,PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Preface

Overview

This document mainly introduces the usage of Rockchip Linux RKNN Demo, aiming to help engineers get started with RKNN Demo development and debugging methods faster.

Product ID

Chipset Name	Buildroot	Debian	Yocto
RK1808	Y	Y	N
RK3399PRO	Y	Y	N

Application Object

This document (this guide) is intended primarily for the following readers:

Field Application Engineer

Software Development Engineer

Revision History

Date	Version	Author	Change Description
2018-12-08	V0.0.1	lhp	initial version
2019-02-15	V0.0.2	lhp	support 1808 and 3399pro
2019-06-05	V0.0.2	Caesar Wang	add rknn_demo FAQ
2020-03-12	V1.0.0	Caesar Wang	markdown initial version

Content

RKNN Demo Developer Guide

1. Run RKNN_DEMO

1.1 Overview

The rknn_demo module configuration directory is "/ buildroot/ package/ rockchip/ rknn_demo" and the code directory is "/external/rknn_demo". It is mainly used to collect images through USB camera and then send them to NPU for processing and display the results through MiniGUI. The currently supported model is “mobilenet_ssd”.

1.2 Configure in buildroot

The required configurations are enabled in SDK by default, the mainly dependences are RGA and USB camera. If they are not enabled, please go to kernel to check the historical changes of related config. Because RKNN interfaces and model of rk1808 and rk3399pro are different, you can configure according to chip type in the configuration file, mainly basing on BR2_PACKAGE_RK1808 and BR2_PACKAGE_RK3399PRO. When it is rk1808, the value of the macro "NEED_RKNNAPI" used in the code is 0 and the value is 1 when it is rk3399pro.

1.3 NPU Related

The model files have been compiled into the board by default in the SDK. The corresponding file macro and directory are as follows:

```
#define MODEL_NAME "/usr/share/rknn_demo/mobilenet_ssd.rknn"
#define BOX_PRIORS_TXT_PATH "/usr/share/rknn_demo/box_priors.txt"
#define LABEL_NALE_TXT_PATH "/usr/share/rknn_demo/coco_labels_list.txt"
```

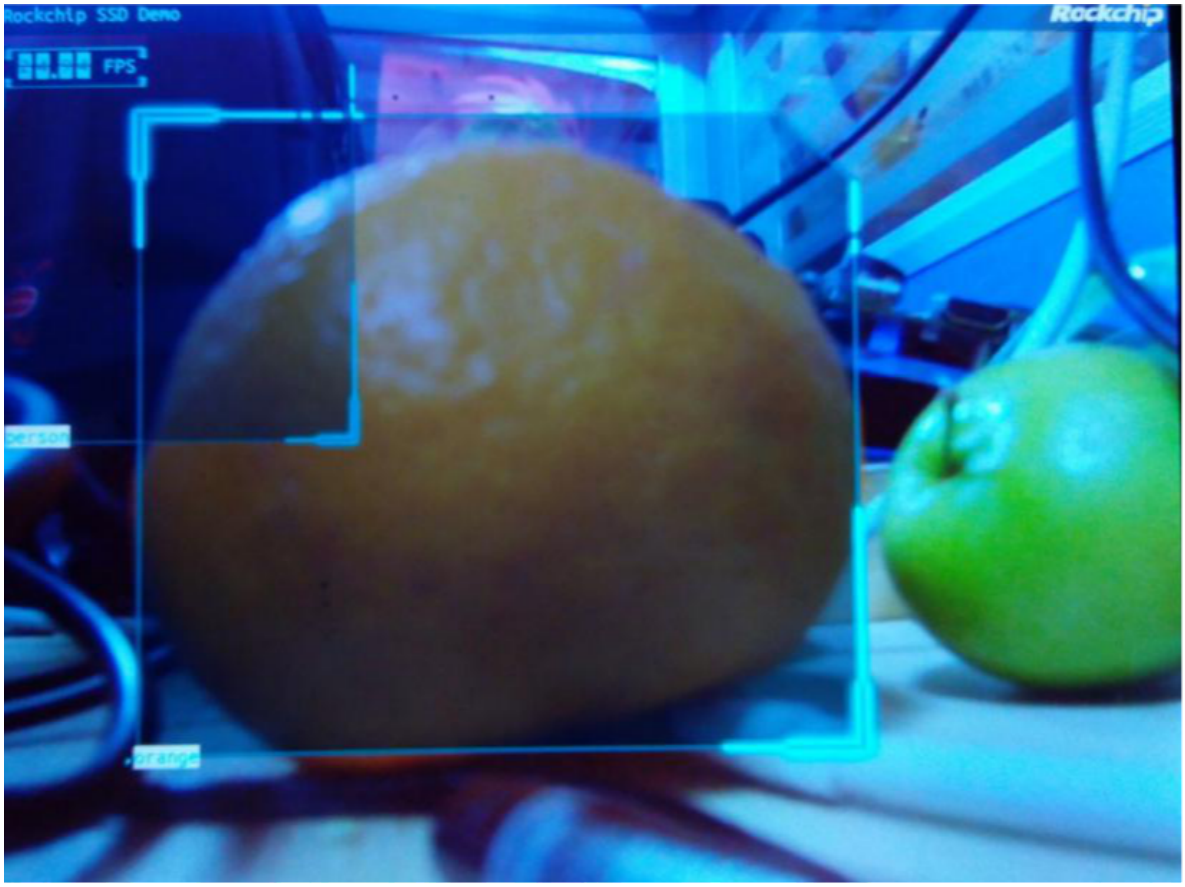
Before the model runs, make sure the related files exist.

1.4 Compile and Run

You can compile modules in the SDK directory with the command “make rknn_demo” and generate the rknn_demo executable file. Before copy to the board, make sure USB camera is plugged in, and run “rknn_demo” command directly.
Note: should not be coexist with other UIs. Please delete the related UI startup commands before starting. The board’s default UI is QT, you can run the command:

```
/etc/init.d/S50launcher stop
```

The normal running frame rate is around 25~30fps. If the frame rate is not enough, the USB camera input frame rate may not be enough. It is recommended to face bright light or replace the USB camera. Unstable connection of USB camera will cause abnormal operation. So please keep a stable connection. The operation result are as follows:



2. RKNN_DEMO Development

2.1 File Directory Introduction

The config.in is a configuration file, the rknn_demo.mk is the basic compilation file in which copy of resource is done.
Detailed commands please refer to RKNN_DEMO_INSTALL_TARGET_CMDS.
The CMakeLists.txt is compiled file in the “src/” code directory. You can add your own files to compile in RKNN_DEMO_SRC.
The rknn_camera.c is the main file which is used to start MiniGUI main window and initialize modules. The MiniGUIMain is the main function entry. The Rknn_ui_show creates a main window for MiniGUI. The rknn_demo_init will start two threads: post and run. Run is used for capturing image and NPU processing, and sends the result to post thread which receives the processing result of NPU and doing post-processing, and outputs the result to display.
The “src/rknn/ssd” is SSD related processing file. In the ssd.c, the ssd_run function loads the model and obtains the buf of USB camera through the cameraRun function, and outputs to the registration function ssd_camera_callback. In the ssd_camera_callback function, yuv_draw function sends video data to MiniGUI layer for RGA synthesis of video data and UI data. YUV420toRGB24_RGA convert video data from 640*480 nv12 format to 300*300 rgb888 format which will be sent to the ssd_rknn_process function for processing.
The src/ui/ssd is UI display file for SSD. The caption_create function paints title bar and displays it in caption_wnd_proc; the fps_create function paints frame rate bar and displays it in fps_wnd_proc; the ssd_paint_object paints region of detected object and the processing result of SSD is sent here for display.
Detailed MiniGUI development and processing, please refer to related open source materials.

3. RKNN_DEMO FAQ

3.1 How to Switch Display 720p Resolution on HDMI

```
[root@rk3399pro:/]# rknn_demo
librga:RGA_GET_VERSION:3.02,3.020000
ctx=0x2607c20,ctx->rgaFd=3
Rga built version:version:+2017-09-28 10:12:42
success build
set plane zpos = 3 (0~3)size = 3686476, g_bo.size = 4259840
size = 3686476, cur_bo->size = 2129920
size = 3686476, cur_bo->size = 2129920
size = 3686476, cur_bo->size = 2129920
NEWGAL: Video mode smaller than requested.
```

In case of the above issue, add the debug information for debugging, as below:

```
external/minigui$ git diff
diff --git a/src/newgal/video.c b/src/newgal/video.c
index f32197a..5641126 100644
--- a/src/newgal/video.c
+++ b/src/newgal/video.c
@@ -524,6 +524,8 @@ GAL_Surface * GAL_SetVideoMode (int width, int height, int
bpp, Uint32 flags)

     GAL_VideoSurface = (mode != NULL) ? mode : prev_mode;

+    GAL_SetError("NEWGAL: mode->w=%d, mode->h=%d, width=%d, height=%d\n",mode-
>w, mode->h, width, height);

Solutions as follows:

(1) How to Switch to a Different Type of Panel
The “/external/minigui” is selected VOP0(VOPB) for display by default. Ensure
that the display device (EDP/HDMI/MIPI..)
is placed on VOPB.

(2) How to Switch Display Resolution
The default resolution is 2048x1536 on RK3399PRO EVB at present. If you need to
switch resolution to 1280x720, the following configuration is needed:
rknn_demo/minigui/MiniGUI-1280x720.cfg and ui/ssd/ssd_ui.c where the resolution
should be changed to 1280x720.
Finally change package/rockchip/rknn_demo/rknn_demo.mk in the buildroot:

```diff
--- a/minigui/MiniGUI-1280x720.cfg
+++ b/minigui/MiniGUI-1280x720.cfg
@@ -48,7 +48,7 @@ defaultmode=800x600-32bpp
 #{{ifdef _MGGAL_SHADOW
 [shadow]
 real_engine=drmcon
-defaultmode=1280x720-16bpp
+defaultmode=720x1280-16bpp
 rotate_screen=ccw
```
```

```
 #}}

diff --git a/ui/ssd/ssd_ui.c b/ui/ssd/ssd_ui.c
index 8e9884d..310e682 100644
--- a/ui/ssd/ssd_ui.c
+++ b/ui/ssd/ssd_ui.c
@@ -15,8 +15,8 @@
 #define DST_W             300
 #define DST_H             300
 #if NEED_RKNNAPI
-#define DISP_W            2048
-#define DISP_H            1536
+#define DISP_W            720
+#define DISP_H            1280
```

Secondly, change the package/rockchip/rknn_demo/rknn_demo.mk in buildroot, as follows:

```
 ifeq ($(BR2_PACKAGE_RK3399PRO),y)
-RKNN_DEMO_MINIGUI_CFG=minigui/MiniGUI-2048x1536.cfg
+RKNN_DEMO_MINIGUI_CFG=minigui/MiniGUI-1280x720.cfg
 endif
```

Lastly, you need rebuild or clean the external/rknn_demo and external/minigui projects.

```
rm buildroot/output/rockchip_rk3399pro_combine/build/rknn_demo-1.0.0/ -rf
rm buildroot/output/rockchip_rk3399pro_combine/build/rknn_demo-1.0.0/ -rf
./build.sh
```