

第2版

画像認識ユニット **NVP-Ax230SDK**
Software Development Kit
Fine Vision Processor
for Linux

OpenCV 利用手順書

maxell
マクセルシステムテック株式会社

はじめに

このたびはNVP-Ax230シリーズの画像処理ユニット(NVP-Ax230CL／235CL)、および、「NVP-Ax230SDK for Linux」をお買い上げいただきまして、誠にありがとうございます。

本マニュアルはNVP-Ax230シリーズを使用したアプリケーション作成のための基本ソフトウェアである「NVP-Ax230SDK for Linux」環境下でOpenCVを利用する方法について記載します。ハードウェア、および、SDKについては各マニュアルをご参照ください。

なお、本マニュアルではNVP-Ax230CL／235CLの画像処理ユニットを特に区別する必要がない場合には、NVP-AX230と記載しています。



- システムの構築やプログラム作成などの操作を行う前に、本マニュアルの記載内容をよく読み、書かれている 指示や注意を十分理解して下さい。誤った操作によりシステムの故障が発生することがあります。
- 本マニュアルの記載内容について理解できない内容、疑問点または不明点がございましたら、弊社営業窓口までお知らせ下さい。また、弊社ホームページのお問い合わせのページからも受け付けてますのでご利用ください。
<http://www.systemtech.maxell.co.jp/solution/vp/>
- お客様の誤った操作に起因する、事故発生や損害につきましては、弊社は責任を負いかねますのでご了承ください。
- 弊社提供のハードウェアおよびソフトウェアを無断で改造しないで下さい。この場合の品質および安全につきましては、弊社は責任を負いかねますのでご了承ください。
- 本マニュアルの内容について予告なく変更する場合があります。

ご注意書き

1. 本ソフトウェアおよび本資料に記載されるこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本ソフトウェアおよび本資料に記載される情報は、正確を期するため慎重に作成したものですが、誤りがないことを保証するものではありません。万一、本ソフトウェアおよび本資料に記載される情報の誤りに起因する損害が生じた場合においても、当社は一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用例等の情報の使用に起因して発生した第三者の特許権、著作権その他知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は本資料に基づき当社または第三者の特許権、著作権その他知的財産権を何ら許諾するものではありません。
4. 本ハードウェアおよび本ソフトウェアを改造、改変、複製等しないでください。かかる改造、改変、複製により生じた損害に関し、当社は一切その責任を負いません。
5. 本ハードウェアおよび本ソフトウェアは、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物質的損害を発生させるおそれのある機器・システム（原子力制御システム・軍事機器等）、もしくは高品質水準が求められる機器・システム（自動車・電車・船舶等の輸送機器、交通用信号機器、防災・防犯装置、各種安全装置等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 本ハードウェアおよび本ソフトウェアをご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、本ハードウェアおよび本ソフトウェアの品質向上に努めておりますが、使用条件によっては誤動作したりする場合があります。本ハードウェアおよび本ソフトウェアのご使用にあたっては、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証、および安全検証を行ってください。
8. 本資料に記載されている当社のハードウェア製品、ソフトウェア製品および技術を国内外の法令および規制により、製造・使用・販売を禁止されている機器・システムに使用することはできません。また、本ハードウェア、本ソフトウェアおよび技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途に使用しないでください。本ハードウェア、本ソフトウェアおよび技術を輸出する場合は「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続きを行ってください。
9. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担頂きますのでご了承ください。
10. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

目次

1. 概要.....	1
1.1 概要	1
1.2 特徴	1
1.3 前提条件	2
2. 提供内容.....	3
2.1 フォルダ構成.....	3
2.2 ファイル一覧	4
2.2.1 cv_vp_sample1フォルダ	4
2.2.2 Documentsフォルダ	4
2.2.3 installフォルダ	4
2.2.4 tutorialsフォルダ	4
3. チュートリアル	5
3.1 環境設定	5
3.2 OpenCVをビルドしない場合	5
3.3 OpenCVビルド手順	6
3.3.1 OpenCVソースコードの入手	6
3.3.2 OpenCVライブラリのビルド	6
3.4 OpenCVチュートリアル	8
3.4.1 OpenCVチュートリアルについて	8
3.4.2 OpenCVチュートリアルのビルド	8
3.4.3 OpenCVチュートリアルの実行	10
3.5 OpenCVとNVP-Ax230SDKの連携	10
3.5.1 cv_vp_sample1のビルド	10
3.5.2 cv_vp_sample1の実行	10
3.5.3 cv_vp_sample1の説明	11
付録A 変更履歴	12
付録B 注意事項	13
付録C OpenCV-2.4.9構築情報	14
付録D OpenCV CMAKE実行ログ	17
付録E OpenCVチュートリアル一覧	20

図・表・リスト目次

図1-1 システム構成	1
図1-2 OpenCV実行例.....	1
図2-1 フォルダー一覧.....	3
図3-1 cv_vp_sample1フローチャート	11
表1-1 Linux開発環境	2
表1-2 NVP-Ax230 for Linux/OpenCV環境.....	2
表2-1 cv_vp_sample1フォルダのファイル一覧.....	4
表2-2 documentsフォルダのファイル一覧	4
表2-3 installフォルダのファイル一覧.....	4
表2-4 tutorialsフォルダのファイル一覧.....	4
表3-1 共有フォルダ	5
表3-2 Ubuntu環境	5
表C-1 OpenCV-2.4.9構築情報(1/3)	14
表C-2 OpenCV-2.4.9構築情報(2/3)	15
表C-3 OpenCV-2.4.9構築情報(3/E).....	16
表E-1 01_core : core module. The Core Functionality	20
表E-2 02_imgproc : imgproc module. Image Processing	21
表E-3 03_highgui : highgui module. High Level GUI and Media.....	22
表E-4 05_feature2d : feature2d module. 2D Features framework	22
表E-5 07_objdetect : objdetect module. Object Detection.....	22
表E-6 08_ml : ml module. Machine Learning	23
表E-7 10_contrib : contrib module. The additional contributions made available !	23
リスト3-1 toolchain.cmake.....	6
リスト3-2 OpenCVチュートリアルのMakefile例.....	9
リストD-1 CMAKE実行ログ(1/3).....	17
リストD-2 CMAKE実行ログ(2/3).....	18
リストD-3 CMAKE実行ログ(3/E)	19

1. 概要

1.1 概要

OpenCV はオープンソースのコンピュータビジョンライブラリです。本ドキュメントは「NVP-Ax230SDK for Linux」環境下で OpenCV を利用する方法について記載します。

1.2 特徴

- NVP-Ax230SDK 提供の画像認識ライブラリと OpenCV ライブラリを同時利用できます。
(お互いの得意な所を利用することでアプリケーション応用の幅が広がります)

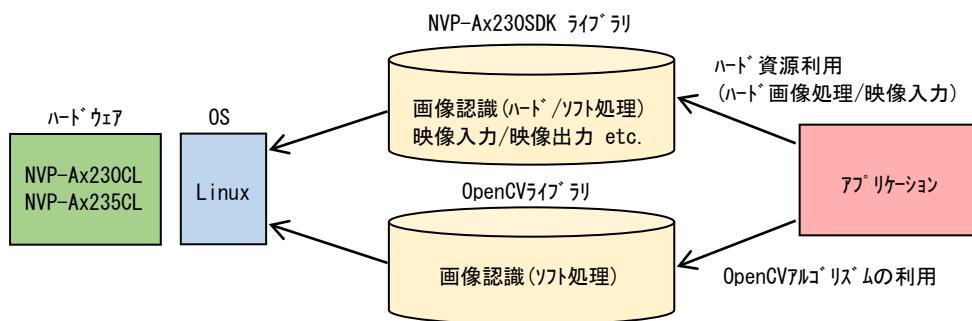


図1-1 システム構成

- OpenCV アプリケーションを組み込みシステム(スタンドアローン)で実行できます(PC 不要)。
- アプリケーション開発は Windows 上の VM から Ubuntu を起動し、クロスコンパイルします。

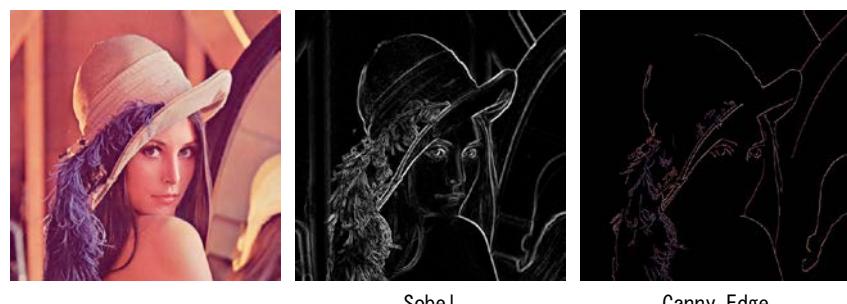


図1-2 OpenCV実行例

1.3 前提条件

前提条件を下記に示します。開発環境の準備、使い方等の詳細は、NVP-Ax230SDK for Linux ユーザーズマニュアルを参照ください。

表1-1 Linux開発環境

項目		内容
ハードウェア	PC	Windows が動作するパソコン 10/100Base T/TX × 1 ポート以上
	NVP-Ax230	NVP-Ax230CL NVP-Ax235CL
OS		Microsoft Windows7 32bit 版 (ServicePack1 以上) (64Bit 版は不可)
VM		ORACLE VM VirtualBox 4.2.10r84104 Ubuntu 12.04
クロスコンパイラ		Sourcery G++ Lite 2010q1-202
ソフトウェア開発キット		NVP-Ax230SDK + NVP-Ax230SDK for Linux

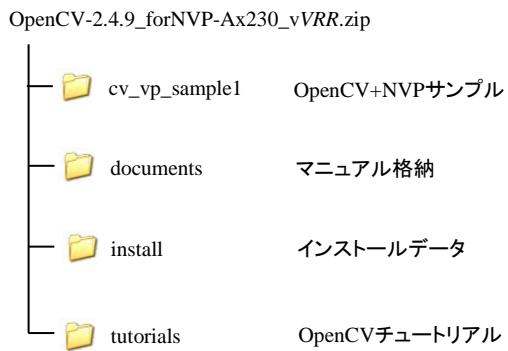
表1-2 NVP-Ax230 for Linux / OpenCV環境

項目	内容
ベース Linux	u-boot 2011.03 Linux 2.6.35.9 (MontaVista Linux 6)
ルートファイルシステム	cramfs (BusyBox v1.13.2 等を含む)
NVP-Ax230SDK ライブライ	libimp.so
OpenCV	OpenCV 2.4.9

2. 提供内容

2.1 フォルダ構成

「NVP-Ax230SDK for Linux」上で動作する OpenCV ライブラリ、および、サンプルプログラムを zip ファイルで提供します。zip ファイル展開後のフォルダ構成を図 2-1 に示します。



VRR: バージョン・リビジョン

図2-1 フォルダ一覧

2.2 ファイル一覧

2.2.1 cv_vp_sample1 フォルダ

cv_vp_sample1 フォルダに格納されるファイル一覧を表 2-2 に示します。

表2-1 cv_vp_sample1 フォルダのファイル一覧

ファイル名	内容
.cv_vp_sample1\img	LinuxLogo.jpg
	WindowsLogo.jpg
.cv_vp_sample1\obj	-
.cv_vp_sample1\src	main.cpp
.cv_vp_sample1	Makefile
	sample1

2.2.2 Documents フォルダ

Documents フォルダに格納されるファイル一覧を表 2-2 に示します。

表2-2 documents フォルダのファイル一覧

ファイル名	内容
.documents	OpenCV 利用手順書_Rev*.pdf

*: ドキュメントバージョン

2.2.3 install フォルダ

install フォルダに格納されるファイル一覧を表 2-3 に示します。

表2-3 install フォルダのファイル一覧

ファイル名	内容
.install	install.tar.gz
	install_no-symlink.tar.gz
	toolchain.cmake

(*) シンボリックリンクが利用可能な環境で利用します。

(**) シンボリックリンクが利用不可能な環境で利用します。

2.2.4 tutorials フォルダ

tutorials フォルダに格納されるファイル一覧を表 2-4 に示します。

表2-4 tutorials フォルダのファイル一覧

ファイル名	内容
.tutorials	tutorials.tar.gz

3. チュートリアル

3.1 環境設定

NVP-Ax230SDK for Linux 上で OpenCV を利用したプログラムを実行する際、共有フォルダを利用します。共有フォルダ設定方法の詳細は NVP-Ax230SDK for Linux ユーザーズマニュアルを参照ください。

表3-1 共有フォルダ

OS	フォルダ名称	備考
Windows7	C:\share	
Ubuntu	/mnt/share	
NVP-Ax230	/media/net	

NVP-Ax230SDK for Linux 上で動作するプログラムを開発する環境として Ubuntu を利用します。Ubuntu のインストール等の事前準備については NVP-Ax230SDK for Linux ユーザーズマニュアルを参照ください。

表3-2 Ubuntu環境

項目	格納場所	備考
クロスコンパイラ	\$HOME/work/tools/arm-2010q1/	Sourcery G++ Lite 2010q1-202
cramfs ファイルシステム	\$HOME/work/rootfs/cramfs	
NVP-Ax230SDK for Linux	/mnt/share/VP230SDK	

3.2 OpenCV をビルドしない場合

OpenCV をビルドしない場合、OpenCV を利用したプログラムをコンパイルするための環境生成が必要となります。OpenCV をビルドする場合は3.3章の作業を行うことで環境生成されます。

提供する `install_no-symlnk.tar.gz` ファイルを `C:\share` フォルダに格納します。OpenCV を利用したプログラムのクロスコンパイル環境を Ubuntu に構築します。

```
ubuntu $ cd /mnt/share/
ubuntu $ mkdir opencv
ubuntu $ cd /mnt/share/opencv
ubuntu $ tar zxvf /mnt/share/install_no-symlnk.tar.gz
```

3.3 OpenCV ビルド手順

3.3.1 OpenCV ソースコードの入手

OpenCV ソースコードを下記 URL から入手します。本チュートリアルでは OpenCV-2.4.9 を前提に説明します。バージョン選定はユーザ任意です。入手ファイル(opencv-2.4.9.zip)は C:\share フォルダに格納します。

<http://sourceforge.net/projects/opencvlibrary/files/opencv-unix/2.4.9/>

3.3.2 OpenCV ライブラリのビルド

(1) Makefile を自動生成するため、cmake を Ubuntu にインストールします。

```
ubuntu $ sudo apt-get install cmake cmake-curses-gui
```

(2) クロスコンパイラを環境変数 PATH に追加します(ユーザ任意)。

```
ubuntu $ export PATH=$WORK/tools/arm-2010q1/bin:$PATH
```

(3) OpenCV ソースコードを展開します。

```
ubuntu $ cd $HOME/work
ubuntu $ mkdir opencv
ubuntu $ cd $HOME/work/opencv
ubuntu $ unzip /mnt/share/opencv-2.4.9
ubuntu $ cd $HOME/work/opencv/opencv-2.4.9
```

(4) toolchain.cmake ファイルを作成します(cmake 用データとして toolchain.cmake ファイルを提供します)。

```
ubuntu $ mkdir build
ubuntu $ cd $HOME/work/opencv/opencv-2.4.9/build
ubuntu $ vi toolchain.cmake
```

リスト3-1 toolchain.cmake

```
set( CMAKE_SYSTEM_NAME Linux )
set( CMAKE_SYSTEM_PROCESSOR arm )
set( CMAKE_C_COMPILER arm-none-linux-gnueabi-gcc )
set( CMAKE_CXX_COMPILER arm-none-linux-gnueabi-g++ )
set( CMAKE_FIND_ROOT_PATH ~/work/rootfs/cramfs )
set( $CMAKE_C_FLAGS "-mcpu=cortex-a9 -O3 -mfloating-abi=hard
                     -fno-tree-vectorize -fno-tree-vectorizer-verbose=9" CACHE STRING "c flags" )
```

※ CMAKE_FIND_ROOT_PATH には cramfs 展開フォルダを指定してください(ユーザ任意)。

(5) Makefile を作成します。下記、コマンド実行するとOpenCV 構築情報の設定を行います。OpenCV 構築情報の詳細は「付録 C OpenCV-2.4.9 構築情報」を参照ください。cmake 実行ログを「付録 D OpenCV CMAKE 実行ログ」に示します。

```
ubuntu $ ccmake -DCMAKE_TOOLCHAIN_FILE=toolchain.cmake ..  
構築情報の選択 (GUI 操作)  
ubuntu $ cmake ..
```

(6) ライブラリ作成とインストールを行います。インストールは\$HOME/work/opencv/opencv-2.4.9/build/install フォルダ以下に bin, include, lib, share が作成されます。

```
ubuntu $ make  
ubuntu $ sudo make install
```

(7) \$HOME/work/opencv/opencv-2.4.9/build/install フォルダを/mnt/share/opencv にコピーします。

```
ubuntu $ mkdir /mnt/share/opencv  
ubuntu $ sudo cp -r install /mnt/share/opencv
```

共有フォルダにコピーすると、シンボリックリンクファイルが作成できない旨、エラーが発生します。恐れ入りますが、/mnt/share/opencv/install/lib フォルダにコピーされたファイル libopencv_xxxx.so.2.4.9 から libopencv_xxxx.so.2.4 と libopencv_xxxx.so を作成してください(下記、実行例)。

```
ubuntu $ cd /mnt/share/opencv/install  
ubuntu $ cp libopencv_xxxx.so.2.4.9 libopencv_xxxx.so.2.4  
ubuntu $ cp libopencv_xxxx.so.2.4.9 libopencv_xxxx.so  
(以下、全ての libopencv_xxxx.so.2.4.9 ファイルについて繰り返す)
```

3.4 OpenCV チュートリアル

3.4.1 OpenCV チュートリアルについて

OpenCV チュートリアルの詳細は下記 URL を参照ください。

<http://sourceforge.net/projects/opencvlibrary/files/opencv-unix/2.4.9/>

提供するチュートリアルの一覧を「付録 E」に示します。提供するチュートリアルは一部、NVP-Ax230 用に変更を加えたものです。以下、/mnt/share/opencv/tutorials/01_core/04_adding_two_images を例に説明します。

3.4.2 OpenCV チュートリルのビルド

(1) 提供する tutorials.tar.gz ファイルを C:\share フォルダに格納します。

(2) tutorials.tar.gz ファイルを /mnt/share/opencv フォルダに解凍します。

```
ubuntu $ cd /mnt/share/opencv/
ubuntu $ tar zxvf /mnt/share/tutorials.tar.gz
```

(3) ビルドするチュートリアルのフォルダに移動します。

```
ubuntu $ cd /mnt/share/opencv/tutorials/01_core/04_adding_two_images
```

(3) Makefile を確認します。必要に応じて Makefile を修正してご利用ください。

- ・ CVINC,CVLIB で指定するパスに OpenCV がインストールされているか?
- ・ CC で指定するクロスコンパイラのパスが正しいか?

リスト3-2 OpenCVチュートリアルのMakefile例

```
# オブジェクトディレクトリ
SRCDIR      =
OBJDIR      =

# 最終ターゲット
TARGET      = adding_two_images

# インクルードファイル
CVINC       = /mnt/share/opencv/install/include

# ライブラリファイル
CVLIB       = /mnt/share/opencv/install/lib

# コンパイル条件等
CC          = /home/renesas/work/tools/arm-2010q1/bin/arm-none-linux-gnueabi-gcc
CCFLAGS     = -I$(CVINC) -mtune=cortex-a9 -march=armv7-a -mhard-float -mfloat-abi=softfp
            -mfpu=neon -ftree-vectorize -O2
LDFLAGS     = -L$(CVLIB)

OBJS        = $(OBJDIR)/adding_two_images.o

#####
# 最終ターゲット
all : $(TARGET)

# 共有ライブラリのビルド
$(TARGET) : $(OBJS)
    @echo "$(CC) $(LDFLAGS) -o $@ ..."
    @$(CC) $(LDFLAGS) -o $@ $^ -lopencv_highgui -lopencv_imgproc -lopencv_core

# 各ファイルのビルド
$(OBJDIR)/adding_two_images.o : $(SRCDIR)/adding_two_images.cpp
    $(CC) $(CCFLAGS) -o $(OBJDIR)/adding_two_images.o -c $(SRCDIR)/adding_two_images.cpp

$(OBJDIR) :
    mkdir $(OBJDIR)

# 生成ファイルの削除
clean :
    rm $(OBJDIR)/*.o
    @if [ -f $(TARGET) ]; then rm -rf $(TARGET); fi
```

(4) make を実行します。クロスコンパイル実行により実行形式ファイルが作成されます。

```
ubuntu $ make
```

3.4.3 OpenCV チュートリアルの実行

(1) チュートリアル実行は NVP-Ax230 上で行います。下記の事前準備が必要です。詳細は NVP-Ax230SDK for Linux ユーザーズマニュアルを参照ください。

- NVP-Ax230 の Linux フォーマット
- NVP-Ax230 の Linux 起動
- ネットワーク設定と共有フォルダのマウント

(2) opencv 共有ライブラリのパスを設定します。

```
NVP-Ax230 $ export LD_LIBRARY_PATH=/media/net/opencv/install/lib
```

(3) ubuntu で make したチュートリアルを実行します。

```
NVP-Ax230 $ cd /media/net/opencv/tutorials/01_core/04_adding_two_images
NVP-Ax230 $ ./adding_two_images
Simple Linear Blender
-----
* Enter alpha [0-1]: 0.5
```

(4) チュートリアル実行結果として、01_04.jpg ファイルが生成されます(Windows からビューアーで確認)。

3.5 OpenCV と NVP-Ax230SDK の連携

3.5.1 cv_vp_sample1 のビルド

(1) 提供する cv_vp_sample1 フォルダを C:\share\opencv フォルダに格納します。

(2) サンプルのフォルダに移動して、make を実行します。クロスコンパイル実行により実行形式ファイルが作成されます。

```
ubuntu $ cd /mnt/share/opencv/cv_vp_sample1
```

3.5.2 cv_vp_sample1 の実行

(1) cv_vp_sample1 の実行は NVP-Ax230 上で行います。結果はディスプレイに表示されます。

```
NVP-Ax230 $ cd /media/net/opencv/cv_vp_sample1
NVP-Ax230 $ ./sample1
```

3.5.3 cv_vp_sample1 の説明

cv_vp_sample1 のフローチャートを以下に示します。OpenCV のメモリ Mat と NVP-Ax230SDK の画像メモリのコピー処理を行うことで、OpenCV の画像処理結果を NVP-Ax230 側で扱っております。

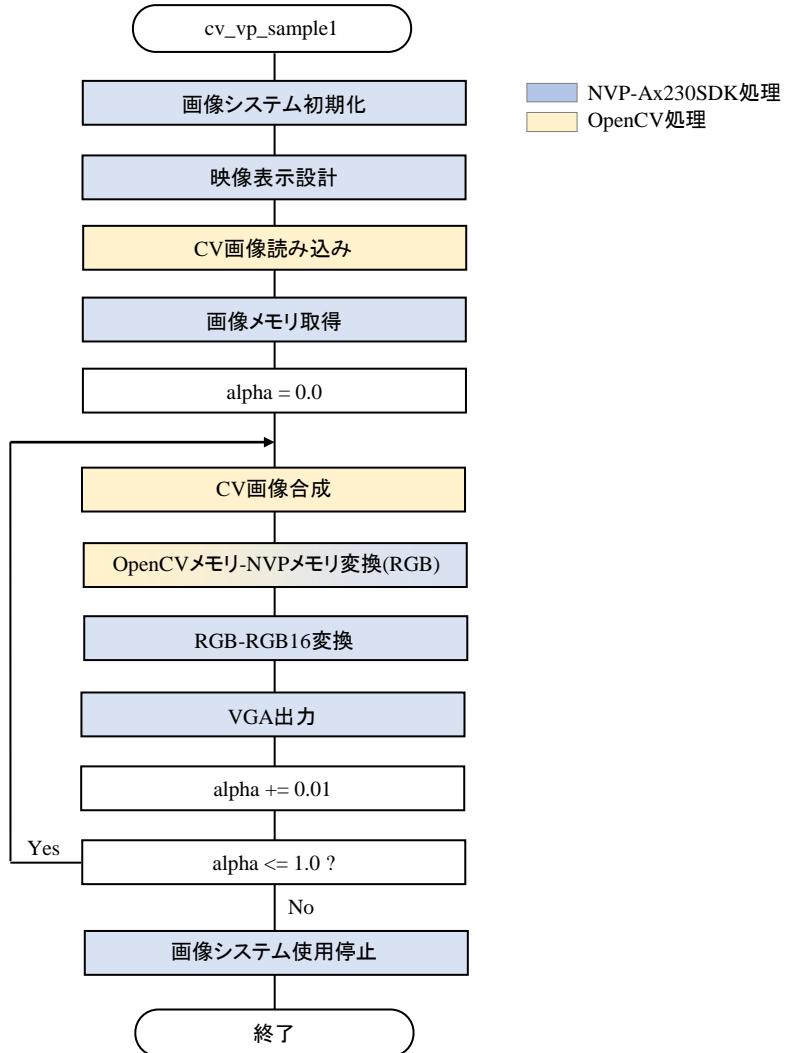


図3-1 cv_vp_sample1フローチャート

付録A 変更履歴

【第1版】

初版

【第2版】

社名を変更しました。

付録B 注意事項

- (1) OpenCV は BSD ライセンスに基づき、商用/非商用問わず無償で使用することができます。
ただし、商用で使う場合は、一部、処理アルゴリズムにおいては、研究用途に限られるものもあるので、ご注意ください。
- (2) 弊社は、配布する NVP-Ax230 用に構築した OpenCV バイナリについて品質保証するものではございません。ご利用に際してはユーザ責任の元、充分評価した上でご利用頂きます様お願いします。
- (3) OpenCV-2.4.9 以外のバージョン利用は、ユーザ責任の元、構築から実施頂く様お願いします。
- (4) 「付録 C OpenCV-2.4.9 構築情報」に記載の通り、GUI、Video I/O 等は無効で構築しております。構築情報を変更してのビルドはユーザ責任の元、実施頂く様お願いします。また、それらの構築情報変更に伴い、カーネル再構築、ルートファイルシステムの変更が必要となる場合もございますが、ユーザ責任の元、実施頂く様お願いします。
- (5) Linux SDK 環境での映像出力(VGA 出力)は、システム起動直後は、Linux の管理下で初期化されます。
du_init サービスコールを発行することで、NVP-Ax230SDK 管理の映像出力機能を有効にすることができますが、以後、Linux 管理下での映像出力は利用不可となります(再度、Linux 管理下での映像出力を実行するためには、一度電源を OFF して再起動する必要があります)ので、ご注意ください。

以下の OpenCV チュートリアルでは Linux 管理下で映像出力をています。また、cv_vp_sample1 は、NVP-Ax230SDK 管理の映像出力機能を利用します。cv_vp_sample1 実行後、以下の OpenCV サンプルは実行しても映像出力が正しく実行されませんのでご注意ください。
 - 01_core/06_2_Drawing2
 - 02_imgproc/01_Smoothing
- (6) 共有ライブラリ(/mnt/share/opencv/install/lib)を NVP-Ax230 から利用するため、cramfs ルートファイルシステムに入れることは論理的に可能ですが cramfs のサイズ制限超過^(*)1)となります。そのため、共有フォルダに格納した上で、環境変数 LD_LIBRARY_PATH で指定する方法を記載しております。

^(*)1) cramfs サイズの上限は 16M バイトです。cramfs 仕様だけでなく、NVP-Ax230 の Flash メモリマップについても 16M バイトの制限があります。

付録C OpenCV-2.4.9 構築情報

表C-1 OpenCV-2.4.9構築情報(1/3)

項目	設定	備考
\$CMAKE_C_FLAGS	-mcpu=cortex-a9 -O3 -mfloat-abi=hard -ftree-vectorize -ftree-vectorizer-verbose=9	
ANT_EXECUTABLE	ANT_EXECUTABLE-NOTFOUND	
BUILD_DOCS	OFF	初期値 ON
BUILD_EXAMPLES	OFF	
BUILD_JASPER	OFF	
BUILD_JPEG	ON	初期値 OFF
BUILD_OPENEXR	OFF	
BUILD_PACKAGE	ON	
BUILD_PERF_TESTS	ON	
BUILD_PNG	ON	初期値 OFF
BUILD_SHARED_LIBS	ON	
BUILD_TBB	OFF	
BUILD_TESTS	ON	
BUILD_TIFF	ON	初期値 OFF
BUILD_WITH_DEBUG_INFO	ON	
BUILD_ZLIB	OFF	
BUILD_opencv_apps	ON	
BUILD_opencv_calib3d	ON	
BUILD_opencv_contrib	ON	
BUILD_opencv_core	ON	
BUILD_opencv_features2d	ON	
BUILD_opencv_flann	ON	
BUILD_opencv_gpu	ON	
BUILD_opencv_highgui	ON	
BUILD_opencv_imgproc	ON	
BUILD_opencv_legacy	ON	
BUILD_opencv_ml	ON	
BUILD_opencv_nonfree	ON	
BUILD_opencv_objdetect	ON	
BUILD_opencv_ocl	ON	
BUILD_opencv_photo	ON	
BUILD_opencv_stitching	ON	
BUILD_opencv_superres	ON	
BUILD_opencv_ts	ON	
BUILD_opencv_video	ON	
BUILD_opencv_videostab	ON	
BUILD_opencv_world	OFF	
CLAMDBLAS_INCLUDE_DIR	CLAMDBLAS_INCLUDE_DIR-NOTFOUND	
CLAMDBLAS_ROOT_DIR	CLAMDBLAS_ROOT_DIR-NOTFOUND	
CLAMDFFT_INCLUDE_DIR	CLAMDFFT_INCLUDE_DIR-NOTFOUND	
CLAMDFFT_ROOT_DIR	CLAMDFFT_ROOT_DIR-NOTFOUND	

表C-2 OpenCV-2.4.9構築情報(2/3)

項目	設定	備考
CMAKE_BUILD_TYPE		
CMAKE_CONFIGURATION_TYPES	Debug;Release	
CMAKE_INSTALL_PREFIX	/home/renesas/work/opencv/opencv-2.4.9/build/install	
CMAKE_VERBOSE	OFF	
CUDA_BUILD_CUBIN	OFF	
CUDA_BUILD_EMULATION	OFF	
CUDA_HOST_COMPILER	/home/renesas/work/tools/arm-2010q1/bin/ arm-none-linux-gnueabi-gcc	
CUDA_SDK_ROOT_DIR	CUDA_SDK_ROOT_DIR-NOTFOUND	
CUDA_SEPARABLE_COMPILATION	OFF	
CUDA_TOOLKIT_ROOT_DIR	CUDA_TOOLKIT_ROOT_DIR-NOTFOUND	
CUDA_VERBOSE_BUILD	OFF	
EIGEN_INCLUDE_PATH	EIGEN_INCLUDE_PATH-NOTFOUND	
ENABLE_COVERAGE	OFF	
ENABLE_NEON	ON	初期値 OFF
ENABLE_NOISY_WARNINGS	OFF	
ENABLE OMIT_FRAME_POINTER	ON	
ENABLE_PRECOMPILED_HEADERS	ON	
ENABLE_PROFILING	OFF	
ENABLE SOLUTION FOLDERS	OFF	
ENABLE_VFPV3	ON	初期値 OFF
EXECUTABLE_OUTPUT_PATH	/home/renesas/work/opencv/opencv-2.4.9/build/bin	
GIGEAPI_INCLUDE_PATH	GIGEAPI_INCLUDE_PATH-NOTFOUND	
GIGEAPI_LIBRARIES	GIGEAPI_LIBRARIES-NOTFOUND	
INSTALL_CREATE_DISTRIB	OFF	
INSTALL_C_EXAMPLES	ON	初期値 OFF
INSTALL PYTHON EXAMPLES	OFF	
INSTALL_TESTS	ON	初期値 OFF
INSTALL_TO_MANGLED_PATHS	OFF	
LIBRARY_OUTPUT_PATH_ROOT	/home/renesas/work/opencv/opencv-2.4.9/build	
OPENCV_CONFIG_FILE_INCLUDE_DIR	/home/renesas/work/opencv/opencv-2.4.9/build	
OPENCV_EXTRA_MODULES_PATH		
OPENCV_WARNINGS ARE_ERRORS	OFF	
OPENEXR_INCLUDE_PATH	OPENEXR_INCLUDE_PATH-NOTFOUND	
PVAPI_INCLUDE_PATH	PVAPI_INCLUDE_PATH-NOTFOUND	
PYTHON_PACKAGES_PATH	lib/python2.7/dist-packages	
SPHINX_BUILD	SPHINX_BUILD-NOTFOUND	
WITH_1394	ON	
WITH_CUBLAS	OFF	
WITH_CUDA	ON	
WITH_CUFFT	ON	
WITH_EIGEN	ON	
WITH_FFMPEG	ON	
WITH_GIGEAPI	ON	
WITH_GSTREAMER	ON	
WITH_GTK	OFF	初期値 ON
WITH_JASPER	ON	
WITH_JPEG	ON	

表C-3 OpenCV-2.4.9構築情報(3/E)

項目	設定	備考
WITH_LIBV4L	OFF	初期値 ON
WITH_NVCUVID	OFF	
WITH_OPENCL	ON	
WITH_OPENCLAMDBLAS	ON	
WITH_OPENCLAMDFFT	ON	
WITH_OPENEXR	ON	
WITH_OPENGL	OFF	
WITH_OPENMP	OFF	
WITH_OPENNI	OFF	
WITH_PNG	ON	
WITH_PVAPI	ON	
WITH_QT	OFF	
WITH_TBB	OFF	
WITH_TIFF	ON	
WITH_UNICAP	OFF	
WITH_V4L	OFF	初期値 ON
WITH_VTK	OFF	
WITH_XIMEA	OFF	
WITH_XINE	OFF	

付録D OpenCV CMAKE 実行ログ

リストD-1 CMAKE実行ログ(1/3)

```
-- Detected version of GNU GCC: 44 (404)
-- Could NOT find ZLIB (missing: ZLIB_LIBRARY ZLIB_INCLUDE_DIR)
-- Could NOT find Jasper (missing: JASPER_LIBRARY JASPER_INCLUDE_DIR)
-- checking for module 'gstreamer-base-0.10'
--   package 'gstreamer-base-0.10' not found
-- checking for module 'libdc1394-2'
--   package 'libdc1394-2' not found
-- checking for module 'libdc1394'
--   package 'libdc1394' not found
-- checking for module 'libavcodec'
--   package 'libavcodec' not found
-- checking for module 'libavformat'
--   package 'libavformat' not found
-- checking for module 'libavutil'
--   package 'libavutil' not found
-- checking for module 'libswscale'
--   package 'libswscale' not found
-- Looking for libavformat/avformat.h
-- Looking for libavformat/avformat.h - not found
-- Looking for ffmpeg/avformat.h
-- Looking for ffmpeg/avformat.h - not found
-- Could NOT find PythonLibs (missing: PYTHON_INCLUDE_DIRS) (Required is at least version "2.7.3")
-- Cannot probe for Python/Numpy support (because we are cross-compiling OpenCV)
-- If you want to enable Python/Numpy support, set the following variables:
--   PYTHON_INCLUDE_PATH
--   PYTHON_LIBRARIES
--   PYTHON_NUMPY_INCLUDE_DIR
-- Could NOT find JNI (missing: JAVA_AWT_LIBRARY JAVA_JVM_LIBRARY JAVA_INCLUDE_PATH
JAVA_INCLUDE_PATH2 JAVA_AWT_INCLUDE_PATH)
-- 
-- General configuration for OpenCV 2.4.9 =====
-- Version control:           unknown
-- 
-- Platform:
--   Host:                  Linux 3.2.0-48-generic-pae i686
--   Target:                Linux arm
--   CMake:                 2.8.7
--   CMake generator:        Unix Makefiles
--   CMake build tool:       /usr/bin/make
--   Configuration:         Release
-- 
-- C/C++:
--   Built as dynamic libs?: YES
--   C++ Compiler:
/home/renesas/work/tools/arm-2010q1/bin/arm-none-linux-gnueabi-g++ (ver 4.4.1)
--   C++ flags (Release):    -fsigned-char -W -Wall -Werror=return-type -Werror=address
-Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations -Wundef
-Winit-self -Wpointer-arith -Wshadow -Wsign-promo -fdiagnostics-show-option -pthread
-fomit-frame-pointer -mfpu=neon -ffunction-sections -O3 -DNDEBUG -DNDEBUG
```

リストD-2 CMAKE実行ログ(2/3)

```
-- C++ flags (Debug): -fsigned-char -W -Wall -Werror=return-type -Werror=address
-Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations -Wundef
-Winit-self -Wpointer-arith -Wshadow -Wsigh-promo -fdiagnostics-show-option -pthread
-fomit-frame-pointer -mfpu=neon -ffunction-sections -g -O0 -DDEBUG -D_DEBUG
-- C Compiler:
/home/renesas/work/tools/arm-2010q1/bin/arm-none-linux-gnueabi-gcc
-- C flags (Release): -fsigned-char -W -Wall -Werror=return-type -Werror=address
-Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations
-Wmissing-prototypes -Wstrict-prototypes -Wundef -Winit-self -Wpointer-arith -Wshadow
-fdiagnostics-show-option -pthread -fomit-frame-pointer -mfpu=neon -ffunction-sections -O3
-DNDEBUG -DNDEBUG
-- C flags (Debug): -fsigned-char -W -Wall -Werror=return-type -Werror=address
-Werror=sequence-point -Wformat -Werror=format-security -Wmissing-declarations
-Wmissing-prototypes -Wstrict-prototypes -Wundef -Winit-self -Wpointer-arith -Wshadow
-fdiagnostics-show-option -pthread -fomit-frame-pointer -mfpu=neon -ffunction-sections -g -O0
-DDEBUG -D_DEBUG
-- Linker flags (Release):
-- Linker flags (Debug):
-- Precompiled headers: YES
--
-- OpenCV modules:
-- To be built: core flann imgproc highgui features2d calib3d ml video legacy
objdetect photo gpu ocl nonfree contrib stitching superres ts videostab
-- Disabled: world
-- Disabled by dependency: -
-- Unavailable: androidcamera dynamicuda java python viz
--
-- GUI:
-- QT: NO
-- GTK+ 2.x: NO
-- GThread: NO
-- GtkGlExt: NO
-- OpenGL support: NO
-- VTK support: NO
--
-- Media I/O:
-- ZLib: zlib (ver 1.2.7)
-- JPEG: libjpeg (ver 62)
-- PNG: build (ver 1.5.12)
-- TIFF: build (ver 42 - 4.0.2)
-- JPEG 2000: build (ver 1.900.1)
-- OpenEXR: build (ver 1.7.1)
--
-- Video I/O:
-- DC1394 1.x: NO
-- DC1394 2.x: NO
-- FFMPEG: NO
-- codec: NO
-- format: NO
-- util: NO
-- swscale: NO
-- gentoo-style: NO
-- GStreamer: NO
-- OpenNI: NO
-- OpenNI PrimeSensor Modules: NO
-- PvAPI: NO
-- GigEVisionSDK: NO
```

リストD-3 CMAKE実行ログ(3/E)

```
-- UniCap:           NO
-- UniCap ucii:      NO
-- V4L/V4L2:          NO/NO
-- XIMEA:             NO
-- Xine:              NO
--
-- Other third-party libraries:
--   Use IPP:          NO
--   Use Eigen:         NO
--   Use TBB:           NO
--   Use OpenMP:        NO
--   Use GCD:           NO
--   Use Concurrency:   NO
--   Use C=:            NO
--   Use Cuda:          NO
--   Use OpenCL:         YES
--
-- OpenCL:
--   Version:          dynamic
--   Include path:     /home/renesas/work/opencv/opencv-2.4.9/3rdparty/include/opencl/1.2
--   Use AMD FFT:      NO
--   Use AMD BLAS:     NO
--
-- Python:
--   Interpreter:      /usr/bin/python (ver 2.7.3)
--
-- Java:
--   ant:               NO
--   JNI:               NO
--   Java tests:        NO
--
-- Tests and samples:
--   Tests:             YES
--   Performance tests: YES
--   C/C++ Examples:   NO
--
-- Install path:        /home/renesas/work/opencv/opencv-2.4.9/build/install
--
-- cvconfig.h is in:   /home/renesas/work/opencv/opencv-2.4.9/build
-----
-- Configuring done
-- Generating done
-- Build files have been written to: /home/renesas/work/opencv/opencv-2.4.9/build
```

付録E OpenCV チュートリアル一覧

(1) 01_core : core module. The Core Functionality

表E-1 01_core : core module. The Core Functionality

項目	実行方法	備考
01_mat_the_basic_image_container	./mat_the_basic_image_container	
02_how_to_scan_images	./how_to_scan_images beautiful_scenery_2560x1600.jpg 50	
03_mat_mask_operations	./mat_mask_operations	
04_adding_two_images	./adding_two_images * Enter alpha [0-1]: <u>0.5</u>	
05_BasicLinearTransforms	./BasicLinearTransforms lena.jpg * Enter the alpha value [1.0-3.0]: <u>2.2</u> * Enter the beta value [0-100]: <u>50</u>	
06_1_Drawing1	./Drawing_1	
06_2_Drawing2	./Drawing_2	VGA 出力
07_discrete_fourier_transform	./discrete_fourier_transform	
08_file_input_output	./file_input_output 01_08.xml ./file_input_output 01_08.yaml	

(2) 02_imgproc : imgproc module. Image Processing

表E-2 02_imgproc : imgproc module. Image Processing

項目	実行方法	備考
01_Smoothing	./Smoothing	VGA 出力
02_Morphology_1	./Morphology_1 Morphology_1_Tutorial_Original_Image.jpg	
03_Morphology_2	./Morphology_2 Morphology_2_Tutorial_Original_Image.jpg	
04_Pyramids	./Pyramids	
05_Threshold	./Threshold chicky_512.png	
06_1_filter2D	./filter2D_demo lena.jpg	
06_2_copyMakeBorder	./copyMakeBorder_demo lena.jpg	
06_3_Sobel	./Sobel_Demo lena.jpg	
06_4_Laplace	./Laplace_Demo Laplace_Operator_Tutorial_Sample_Image.jpg	
06_5_CannyDetector	./CannyDetector_Demo lena.jpg	
06_6_HoughLines	./HoughLines_Demo Hough_Lines_Tutorial_Original_Image.jpg	
06_7_HoughCircle	./HoughCircle_Demo circles.png	
06_8_Remap	./Remap_Demo Remap_Tutorial_Sample_Image.jpg	
06_9_GeometricTransforms	./Geometric_Transforms_Demo Warp_Affine_Tutorial_Sample_Image.jpg	
07_1_EqualizeHist	./EqualizeHist_Demo Histogram_Equalization_Sample_Image.jpg	
07_2_calcHist	./calcHist_Demo Histogram_Calculation_Original_Image.jpg	
07_3_compareHist	./compareHist_Demo Histogram_Comparison_Source_0.jpg Histogram_Comparison_Source_1.jpg Histogram_Comparison_Source_2.jpg	
07_4_calcBackProject	./calcBackProject_Demo1 hand_sample2.jpg	
07_5_MatchTemplate	./MatchTemplate_Demo Template_Matching_Original_Image.jpg Template_Matching_Template_Image.jpg	
08_1_findContours	./findContours_demo HappyFish.jpg	
08_2_hull	./hull_demo Hull_Sample_Image.jpg	
08_3_generalContours1	./generalContours_demo1 Bounding_Rotated_Ellipses_Sample_Image.jpg	
08_4_generalContours2	./generalContours_demo2 Bounding_Rects_Circles_Sample_Image.jpg	
08_5_moments	./moments_demo Moments_Sample_Image.jpg	
08_6_pointPolygonTest	./pointPolygonTest_demo Point_Polygon_Test_Source_Image.png	

(3) 03_highgui : highgui module. High Level GUI and Media

表E-3 03_highgui : highgui module. High Level GUI and Media

項目	実行方法	備考
01_AddingImagesTrackbar	./AddingImagesTrackbar	

(4) 05_feature2d : feature2d module. 2D Features framework

表E-4 05_feature2d : feature2d module. 2D Features framework

項目	実行方法	備考
01_1_cornerHarris	./cornerHarris_Demo Harris_Detector_Sample_Image.jpg	
01_2_goodFeaturesToTrack	./goodFeaturesToTrack_Demo Harris_Detector_Sample_Image.jpg	
01_3_cornerDetector	./cornerDetector_Demo Hough_Lines_Tutorial_Original_Image.jpg	
01_4_cornerSubPix	./cornerSubPix_Demo Corner_Subpixels_Sample_Image.jpg	
02_SURF_detector	./SURF_detector boardPart.jpg board.jpg	
03_SURF_descriptor	./SURF_descriptor boardPart.jpg board.jpg	
04_SURF_FlannMatcher	./SURF_FlannMatcher boardPart.jpg board.jpg	
05_SURF_Homography	./SURF_Homography boardPart.jpg board.jpg	

(5) 07_objdetect : objdetect module. Object Detection

表E-5 07_objdetect : objdetect module. Object Detection

項目	実行方法	備考
01_objectDetection	./objectDetection lena.jpg	
02_objectDetection2	./objectDetection2	

(6) 08_ml : ml module. Machine Learning

表E-6 08_ml : ml module. Machine Learning

項目	実行方法	備考
01_introduction_to_svm	./introduction_to_svms	
02_non_linear_svms	./non_linear_svms	

(6) 10_contrib : contrib module. The additional contributions made available !

表E-7 10_contrib : contrib module. The additional contributions made available !

項目	実行方法	備考
01_retina_tutorial	./retina_tutorial -image lena.jpg	

画像認識ユニット NVP-Ax230SDK for Linux
OpenCV利用手順書（第2版）

(C)マクセルシステムテック株式会社

開発元

マクセルシステムテック株式会社

設計部 〒992-0021 山形県米沢市花沢3091-6

営業部 〒244-0801 神奈川県横浜市戸塚区信濃町549-2三宅ビル

技術サポート窓口 URL <http://www.systemtech.maxell.co.jp/>
mail : vp-support@maxell.co.jp