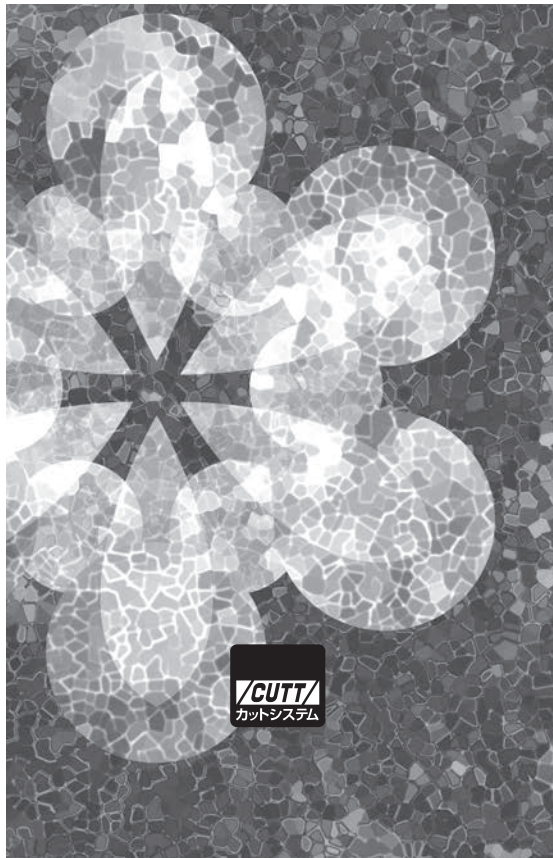


Pythonで始める OpenCV4 プログラミング

北山直洋◎著



■ サンプルファイルのダウンロードについて

本書掲載のサンプルファイルは、一部を除いてインターネット上のダウンロードサービスからダウンロードすることができます。詳しい手順については、本書の巻末にある袋とじの内容をご覧ください。

なお、ダウンロードサービスのご利用にはユーザー登録と袋とじ内に記されている番号が必要です。そのため、本書を中古書店から購入されたり、他者から貸与、譲渡された場合にはサービスをご利用いただけないことがあります。あらかじめご承知おきください。

- 本書の内容についてのご意見、ご質問は、お名前、ご連絡先を明記のうえ、小社出版部宛文書（郵送またはE-mail）でお送りください。
- 電話によるお問い合わせはお受けできません。
- 本書の解説範囲を越える内容のご質問や、本書の内容と無関係なご質問にはお答えできません。
- 匿名のフリーメールアドレスからのお問い合わせには返信しかねます。

本書で取り上げられているシステム名／製品名は、一般に開発各社の登録商標／商品名です。本書では、™ および® マークは明記していません。本書に掲載されている団体／商品に対して、その商標権を侵害する意図は一切ありません。本書で紹介している URL や各サイトの内容は変更される場合があります。

はじめに

OpenCV (Open Source Computer Vision Library) は、膨大な関数を用意した画像処理ライブラリです。一般的な二次元の画像処理、ヒストグラム処理、テンプレートマッチング、オプティカルフロー、およびオブジェクト認識など多様なアプリケーションを開発できる関数群を用意しています。OpenCV を利用すると、数行のコードを記述するだけで簡単な画像処理プログラムを開発できます。このような環境は単に OpenCV だけがもたらしてくれた訳ではなく、ハードウェアやオペレーティングシステムの進歩と相まって提供されています。

OpenCV は当初、C/C++ 言語用に用意されていました。しかし、今では様々な言語から利用できる環境が用意されています。本書は、OpenCV が用意している Python 用のライブラリを使用し、Python から OpenCV を利用する方法を紹介します。Python については、ある程度の知識があることを前提としています。もし、Python をまったく学習していない場合、先に Python の入門書を読むことをお勧めします。

本書は技術書に分類される書籍です。さらに細かく分類すると、専門的な知識が身に付いていない方向けの入門書と言えます。入門書には多くの役割がありますが、個人的に重要な役割は、

- 環境の導入を分かりやすく解説すること
- 技術の魅力を実感させること

の二点だと考えています。

本書は、Anaconda のインストールによる Python での開発環境の実装方法と、Anaconda への OpenCV のインストール方法の解説を掲載しているため、環境の導入は問題なく行えます。

また、OpenCV の基本的な使い方から Deep Learning を用いた画像認識まで解説しているため、技術の魅力を多少ではありながら実感させることができると考えています。

本書に刺激を受けて OpenCV の道を邁進する読者が現れることを期待します。

対象読者

- Python から OpenCV 4.x、OpenCV 3.x を使用したい人
- 画像処理・動画処理プログラミング入門者
- Python と OpenCV の概要を知りたい人

謝辞

出版にあたり、お世話になった株式会社カットシステムの石塚勝敏氏に深く感謝いたします。

2019 年初春 北山直洋

本書の使用にあたって

開発環境、および、実行環境の説明を行います。

■プラットフォーム

OpenCV はマルチプラットフォームをサポートしています。今回は、Windows を開発環境・実行環境とします。

■ Windows バージョン

特に Windows バージョンへの依存はありませんが、開発や動作チェックを行ったシステムは Windows 10 です。

■ Linux 環境

本書は Windows 環境で開発や動作チェックを行っていますが、本書の末尾に付録として、Linux 環境でのプログラム実行方法とインストール方法を解説しています。

■ OpenCV のバージョン

OpenCV 4.0.0 を使用します。OpenCV 3.4.0 や他のバージョンも確認しましたが、最終確認は OpenCV 4.0.0 で行いました。

■ Anaconda と Python のバージョン

Anaconda 5.3.1、Python 3.7 を使用します。Python 2.7 でも確認しましたが、最終確認は Python 3.7 で行いました。

■ Spyder のバージョン

Spyder 3.3.2 を使用します。

■ ユーザーアカウント

最近の Windows ではユーザーやアカウントの管理が強化されています。たとえば、「標準ユーザー」ではプログラムのインストールやアンインストールは制限されます。開発環境やモジュールのインストールやセットアップで警告が出ることがありますので、なるべく「管理者」で実行してください。もちろん、管理者アカウントを使用する場合、危険なこともできるので、十分注意してください。

■ 例外処理

関数の引数に間違った値を指定すると、プログラムが異常終了することがあります。本書は

ごく単純な例外処理を組み込みました。より安全なプログラムとする場合は、自身で異常処理に対応した機構を組み込んでください。また、異常処理への対応を簡略化した理由は、プログラムをシンプルにするためです。

■ソースコードの文字コード

UTF-8 を使用します。

■ URL

URL の記載がありますが、執筆時点のものであり、変更される可能性もあります。リンク先が存在しない場合、キーワードなどから検索してください。

■関数の引数や戻り値の型

2 章～ 10 章に記載している関数の説明において、引数や戻り値に型を記載しており、その型で動作確認をしています。しかし、OpenCV のドキュメントに型が明記されていないことと関係して、本書に記載されている型ではなくても動作することがあります。

用語

用語の使用に関して説明を行います。

■ディレクトリとフォルダ

基本的に「フォルダ」をメインで使用します。「フォルダー」と表記されることも多いですが、最後の「ー」を使用せず「フォルダ」と表記する場合があります。

■画像と配列

画像を保持する場合、NumPy を使用します。これらは基本的に配列を管理します。このため、画像を配列と表現する場合と、その逆の場合もあります。ただし、これらは同じものを指します。

■「/」と「¥」

ソースコードを記述する際に、「/」と「¥」のどちらでも構わない場合、「/」を採用しています。なお、本文中で「¥」が適している場合でも、ソースコードが「/」を使用している場合、「/」で表現しています。

■ソースリストとソースコード

基本的に同じものを指しますが、ソースリストと表現する場合はソース全体を、ソースコードと表現する場合はソースの一部を指す場合が多いです。

■オブジェクト

インスタンスと表現した方が良い場合でも、オブジェクトと表現している場合があります。両方を、厳密に使い分けていませんので、文脈から判断してください。あるいは、オブジェクトと表現した場合、物体を指す場合もありますので文脈から判断してください。

■映像とフレーム

カメラから取得した画像を映像と表現する場合とフレームと表現する場合がありますが、同じものを指します。これらは文脈から判断してください。

■配列やリストについて

配列と表現した場合、タプル、リストそして `numpy.ndarray` などを指しています。このような混在した使用は Python、NumPy、そして OpenCV を同時に説明しているためです。これらの表現が何を指すかは文脈から分かるようにしており、混乱は起きないようにしています。

ライセンス

OpenCV も、一般的なオープンソースに漏れず、下記に示すライセンスが示されています。以降の内容に同意できない場合、OpenCV のダウンロード、コピー、インストール、および使用をしないでください。

ただ、通常のオープンソースのライセンス同様、難しい使用条件は付帯されていません。通常、本ライセンスが OpenCV 使用の制限になることは考えられません。ただ、必ず全文に自身で目を通し、内容を確認して使用してください。OpenCV のライセンスについては、パッケージ内の OpenCV インストールフォルダにある LICENSE に記述されています。

■以下原文（改行などから読みにくいですが、原文をそのまま掲載しています）

By downloading, copying, installing or using the software you agree to this license.
If you do not agree to this license, do not download, install,
copy or use the software.

License Agreement
For Open Source Computer Vision Library
(3-clause BSD License)

Copyright (C) 2000-2018, Intel Corporation, all rights reserved.
Copyright (C) 2009-2011, Willow Garage Inc., all rights reserved.
Copyright (C) 2009-2016, NVIDIA Corporation, all rights reserved.
Copyright (C) 2010-2013, Advanced Micro Devices, Inc., all rights reserved.
Copyright (C) 2015-2016, OpenCV Foundation, all rights reserved.
Copyright (C) 2015-2016, Itseez Inc., all rights reserved.
Third party copyrights are property of their respective owners.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the copyright holders nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall copyright holders or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

なお、本書で扱うソフトウェアにはサードパーティやオープンソースで提供されているものが多数あります。使用において問題が起きそうなものは見当たりませんが、ライセンスが異なる場合がありますので、ライセンスに関しては、ひとつおり目を通してください。

目次

はじめに iii

第1章 開発環境の準備 1

- 1.1 Anaconda のインストール 2
- 1.2 Spyder の起動 8
- 1.3 はじめての Python プログラム 13
- 1.4 OpenCV のインストール 17
- 1.5 OpenCV のインストール確認 21

第2章 はじめての OpenCV プログラム 23

- 2.1 OpenCV とは 24
- 2.2 OpenCV 3、OpenCV 4 の特徴と変更点 25
 - 2.2.1 OpenCV 3 25
 - 2.2.2 OpenCV 4 28
- 2.3 NumPy モジュール 29
- 2.4 画像を生成するプログラム 31
- 2.5 画像の読み込み、書き込みを行うプログラム 38
- 2.6 関数の説明 39

第3章 グラフィックス 43

- 3.1 円を描く 44
- 3.2 画像の上に線を描く 45
- 3.3 画像の上に円を描く 53
- 3.4 画像の上に文字を描く 54
- 3.5 関数の説明 56

第4章	アフィン変換	61
4.1	フリップ.....	62
4.2	リサイズ.....	64
4.3	回転.....	66
4.4	透視投影.....	69
4.5	トリミング.....	73
4.6	関数の説明.....	74
第5章	色の処理など	81
5.1	グレースケール.....	82
5.2	輝度平滑化.....	84
5.3	閾値処理・その1 (スレッシュホールド処理).....	86
5.4	閾値処理・その2 (アダプティブスレッシュホールド処理).....	93
5.5	カラー画像の各成分を分離.....	95
5.6	関数の説明.....	98
第6章	フィルタ処理	103
6.1	画像の色反転.....	104
6.2	ブラー処理.....	105
6.3	メディアン処理.....	106
6.4	ガウシアン処理.....	108
6.5	ラブラシアン処理.....	110
6.6	Sobel 処理.....	111
6.7	Canny 処理.....	112
6.8	画像の膨張.....	113
6.9	画像の収縮.....	115
6.10	ボックスフィルタ処理.....	117
6.11	モザイク処理.....	118
6.12	関数の説明.....	119

第7章 二つの画像合成 129

7.1	二つの画像を加算	130
7.2	二つの画像を減算	143
7.3	二つの画像をブレンド	145
7.4	二つの画像の論理和	146
7.5	関数の説明	148

第8章 動画処理..... 153

8.1	フレームサイズの表示	154
8.2	動画表示	155
8.3	グレイスケール	157
8.4	輝度平滑化	159
8.5	Canny 処理	160
8.6	シャッフル	163
8.7	回転	167
8.8	一部回転	169
8.9	関数の説明	172

第9章 オブジェクト検出 175

9.1	コーナー検出	176
9.2	顔検出	178
9.3	オブジェクト検出	183
9.4	動画の顔検出	185
9.5	オブジェクト除去	188
9.6	ダメージ補修	190
9.7	テンプレートマッチング	194
9.8	特徴点検出	198
9.9	関数の説明	203

第 10 章 Deep Learning 209

10.1	Deep Learning とは	210
10.2	事前準備	213
10.2.1	Chainer のインストール	213
10.2.2	MNIST を用いた学習モデルの作成	216
10.2.3	実行	223
10.3	カメラで撮影した手書き数字の認識	224
10.4	関数の説明	232

付 録 Linux 環境で OpenCV を使用する場合 235

A.1	Linux 環境に関して	236
A.2	本書のプログラムを Linux 環境で動作させる	236
A.3	Ubuntu への Anaconda、OpenCV のインストール	240

	参考文献、参考サイト、参考資料	244
	索引	245

1

開発環境の準備

Python と OpenCV を利用するための環境を整えます。

1.1

Anaconda のインストール

Python を使用できる環境は、様々なものがあります。Python の公式サイト (<https://www.python.org/>) から Python のみをインストールし、必要な機能を実装するために後から個別にライブラリ (パッケージ) をインストールするのも良いでしょう。ここでは、IDE やコンソールなどの Python でよく使用されるライブラリがパッケージされ、広く使われている Anaconda という Python パッケージの最新版を使用します。以降に、Windows へ Anaconda をインストールする手順を示します。

まず、Anaconda for Windows を公式ホームページからダウンロードします。Anaconda インストーラのダウンロードページにアクセスします (<https://www.anaconda.com/download/>)。

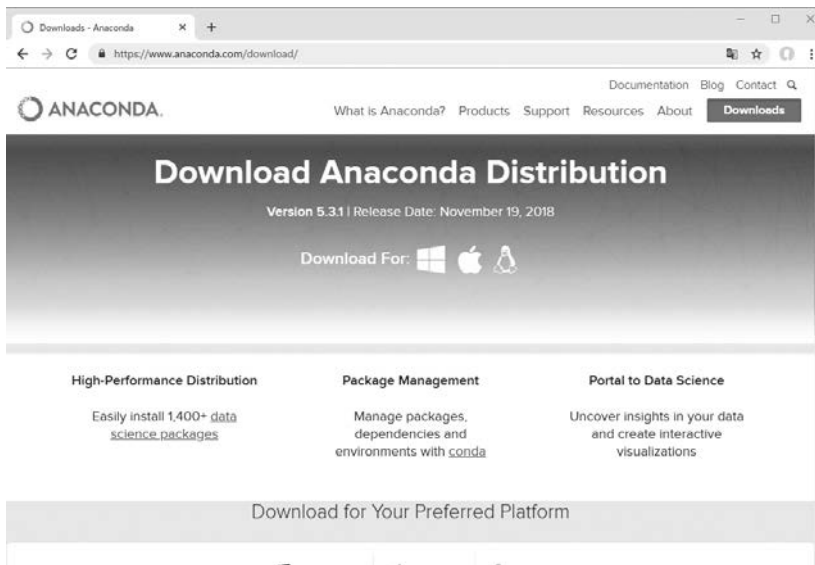


図1.1 ● Anaconda インストーラのダウンロードページ

アクセスすると、図 1.1 のようなページが表示されます。下にスクロールし、「Python 3.x version」の「Download」ボタンをクリックすると 64 ビット用のインストーラがダウンロードされます。「Download」ボタンの下に、64 ビット用と 32 ビット用のインストーラのリンクがあるので、明示的にビット数を選びたい場合は、下のリンクをクリックしてダウンロード

してください。Python 2.7 系の Anaconda も提供されていますので、2.x 系を使用したい場合は、そちらをクリックしてください。本書は、今後のことを考え、最新の Python 3 系をダウンロードします。

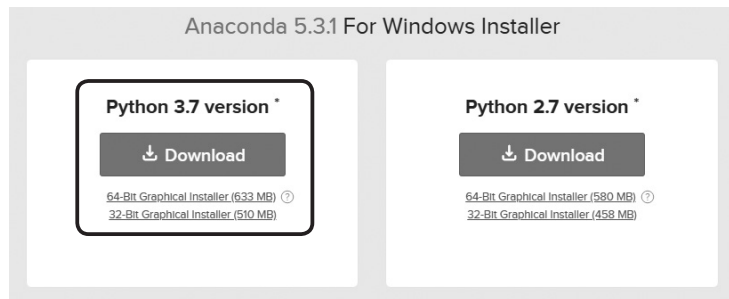


図1.2●Anacondaインストーラのダウンロード

「Download」ボタンをクリックすると、Anaconda の Cheat Sheet を受け取る e-mail アドレス登録の案内が表示されます。必要であれば e-mail アドレスを入力してから「Get the Starter Guide」をクリックし、必要でなければ「No Thanks」をクリックしてください。

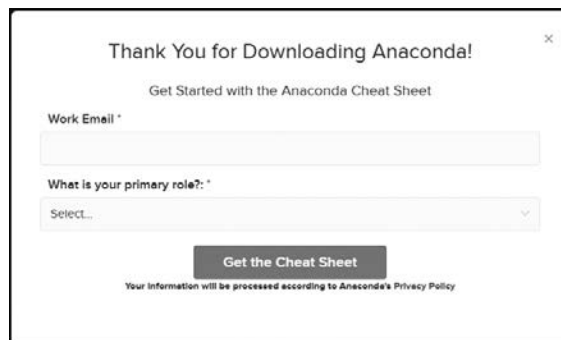


図1.3●e-mailアドレス登録の案内

ダウンロードが完了したら、ブラウザの下部に表示されるインストーラをクリックします。

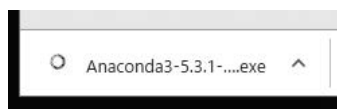


図1.4●インストーラの起動

インストーラが起動すると、Anaconda のバージョンやビット数の表示も行われますので、目的の Anaconda であることを確認し、「Next >」 ボタンをクリックして次の画面に進みます。



図1.5 ● インストーラの起動直後の画面

ライセンスの確認画面が表示されるので、最後までスクロールしてライセンスを一通り確認し、「I Agree」 ボタンをクリックして次の画面に進みます。

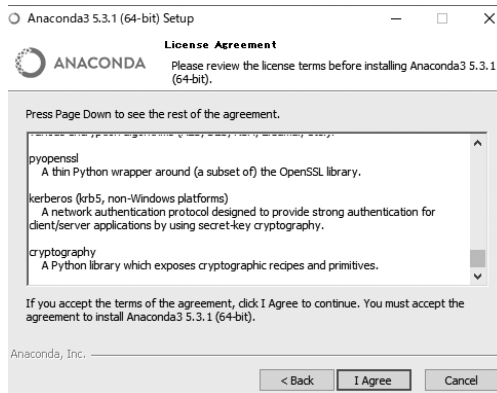


図1.6 ● ライセンスの確認画面

ユーザーとしてインストールするか、全ユーザーへインストールするかを選択する画面が表示されます。ユーザーとして (Just Me) インストールすることが推奨されています。複数の人でパソコンを共用し、全員が同じバージョンの Anaconda を使用するような特殊な環境の場合のみ、「All Users」を選択してください。それ以外の場合は「Just Me」を選択し、「Next >」 ボタンをクリックして次の画面に進みます。

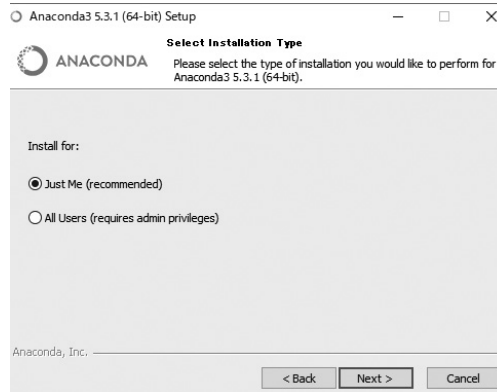


図1.7●インストール対象ユーザーの選択

インストール先を選択する画面が表示されます。特別な理由がない限り変更する必要はないでしょう。インストール先を確認・選択し「Next >」ボタンをクリックして、次の画面に進みます。c:\Users の配下にインストールしたくない場合は、「Browse…」ボタンをクリックし、適切な場所へインストールしてください。

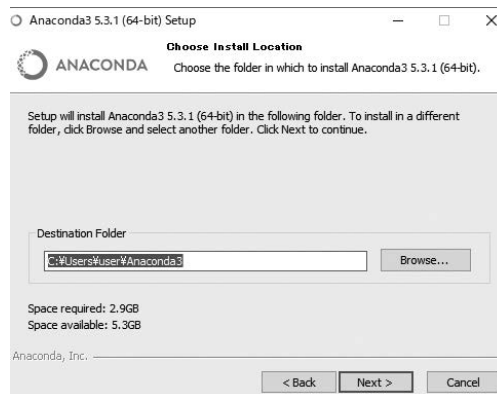


図1.8●インストール先の選択

インストールのオプションを選択する画面が表示されます。特に変更の必要はありませんので、このまま「Install」ボタンをクリックし、インストールを開始します。「Add Anaconda to my PATH environment variable」は、環境変数 PATH に Anaconda のフォルダを追加するか否かの選択肢です。最初の方に「Not recommended…」と記述されており推奨されていません。これをチェックしなくても、コンソールを使用したいときは「Anaconda Prompt」を使用すると Python へのパスは通っています。また、「Register Anaconda as my default

Python 3.7」は Anaconda をデフォルトの Python 3.7 として登録するか否かの選択肢です。これにチェックを付けておくと、インストールした Python がシステム上のプライマリとして扱われますので、ほかの開発ツール、例えば Visual Studio などが、この Python (Anaconda) を自動で認識します。詳細は表示されているメッセージを参照してください。

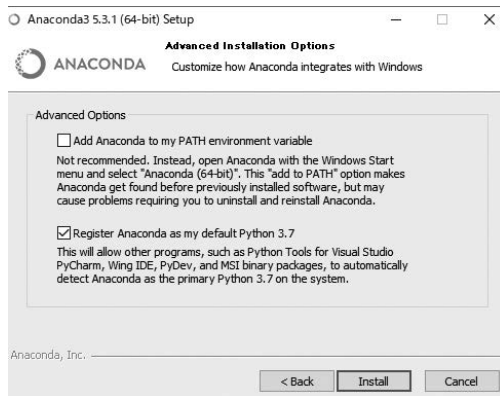


図1.9●インストールのオプションの選択

しばらくインストール作業が続くので、完了を待ちます。

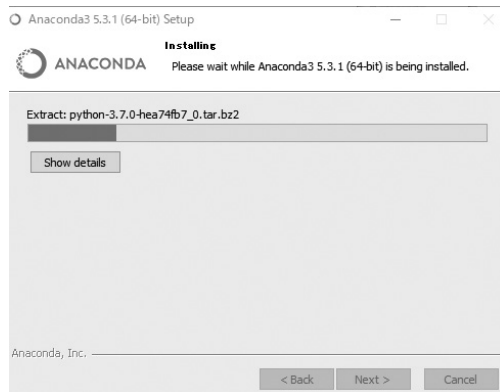


図1.10●インストール中の画面

インストールが完了すると、次のように「Completed」という表示に変わります。「Next >」をクリックして次の画面に進みます。示した画面は「Show details」ボタンをクリックした状態です。クリックしないと、インストール状況は表示されません。

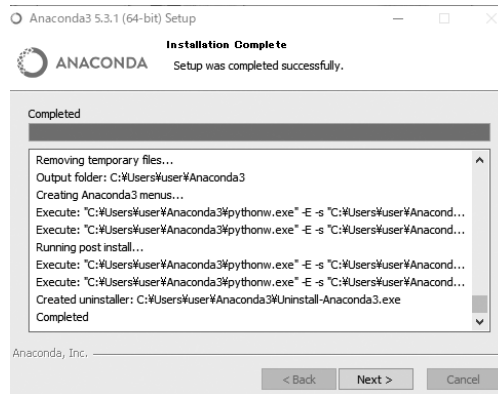


図1.11 ● インストール完了時の画面

コードエディターである Visual Studio Code のインストール確認を行う画面が表示されます。本書では、Visual Studio Code を使用しないため、「Skip」ボタンをクリックして次の画面に進みます。

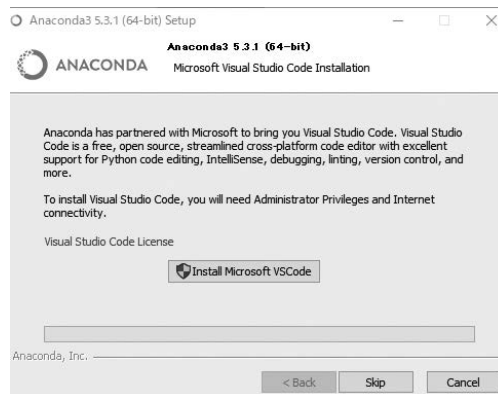


図1.12 ● Visual Studio Code のインストール確認

インストール作業が終わったことを表す画面が表示されます。「Finish」ボタンをクリックし、インストーラを終了させます。二つのチェックボックスにチェックしているので、学習用の画面が立ち上がりますが、必要なければ閉じてください。いずれにしても Anaconda はインストールされています。

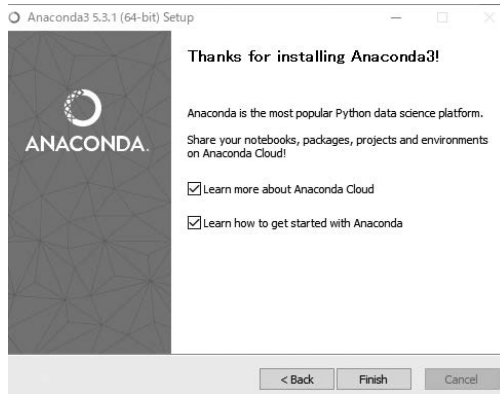


図1.13 ● インストーラの終了

1.2

Spyder の起動

ここでは、Anaconda にパッケージされている IDE の Spyder を起動する方法を説明します。

Anaconda のインストールが完了すると、スタートメニューに Anaconda のフォルダが追加されます。この中に各種ツールを起動できる Anaconda Navigator というソフトウェアがあるので、Anaconda Navigator を起動しましょう。Anaconda Navigator は、Anaconda のフォルダだけでなく、スタートメニューの「最近追加されたもの」にも表示されるので、そちらから起動しても構いません。

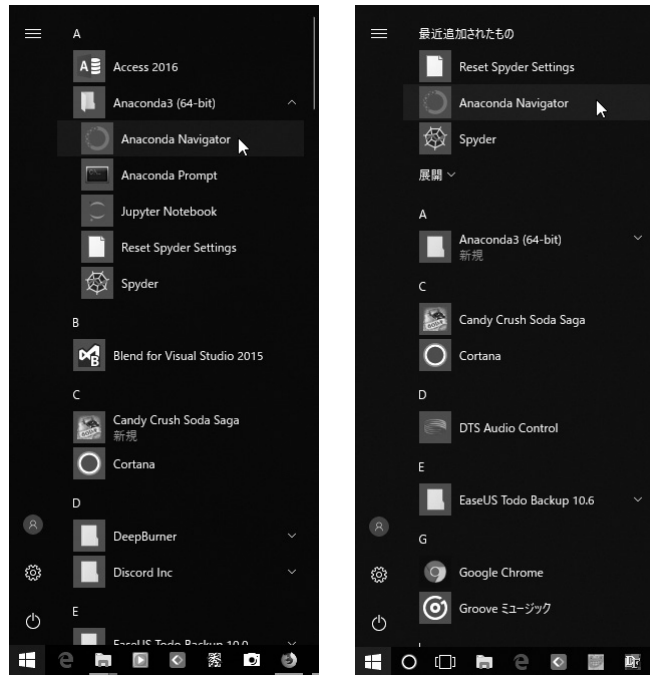


図1.14 ● Anaconda Navigatorの起動

最初の実行で、Anaconda の改善に協力するか否かを求められます。ここでは、何も変更せず「OK, and don't show again」ボタンをクリックします。つまり、改善に協力し、この画面を次回以降表示させないようにします。メッセージにあるように、この設定はいつでも変更可能です。

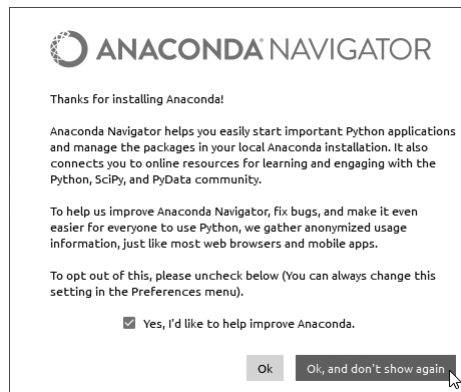


図1.15 ● Anacondaの改善に協力するか否かの確認

Anaconda Navigator が起動すると、図 1.16 に示す画面が現れます。左側に、Environments、Learning、Community や Documentation などが存在します。これらはとても有益ですが、本題からそれるので、ここでは説明を割愛します。

Spyder の欄にある「Launch」ボタンをクリックし、Spyder を起動します。

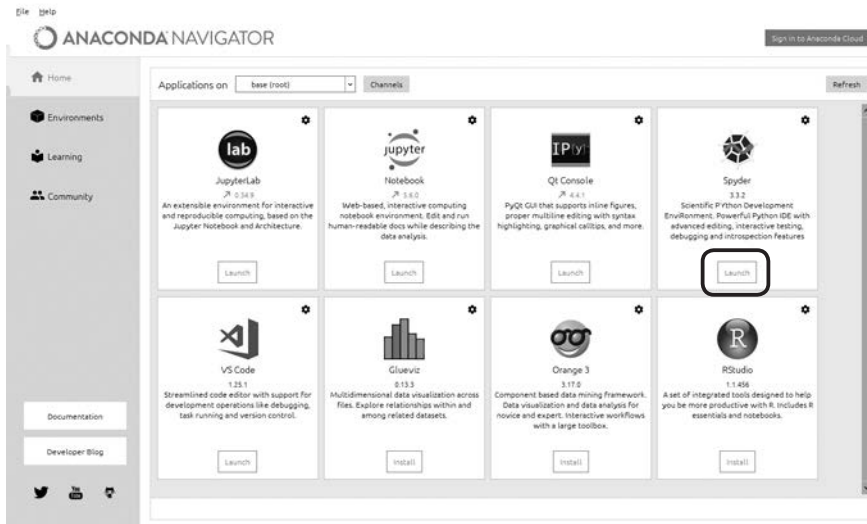


図1.16●Anaconda Navigatorの画面

しばらくすると Spyder が起動します。起動前に、ファイアウォールが警告を発する場合がありますので、そのような場合は通信を許可してください。