

スクリプトなのに Python 計算が高速な理由

ご購入はこちら

西住 流



(a) 元画像



(b) 2値化画像

図1 画像の2値化処理を例にPython (特に数値計算ライブラリ NumPy) の高速さを確かめてみる

Python 計算が高速な理由

● その1: そもそも計算ライブラリの実体はC記述

一般に、スクリプト言語は簡便に利用できる反面、C言語などの高級言語と比較して処理速度が遅くなる問題を抱えています。

Pythonも例外ではなく、標準ライブラリ、関数、構文を使って大量のデータを処理すると、速度が著しく低下します。

しかし、数値計算ライブラリ NumPy を使って処理を実装すると、C言語と同等の処理速度を出すことができます。NumPyのAPI内部がC言語やFortranで実装されているためです。

● その2: とにかく NumPy が書きやすく高速

NumPyは、Pythonで数値計算をする上で必須といえる数値計算ライブラリです。

ndarrayと呼ばれるN次元配列のオブジェクトを使って、少ないコード量で効率良く、かつ高速に数値計算ができます(表1)。

表1 数値計算ライブラリ NumPy を使うと実行時間を何十倍も短くできる

記述方法	実行時間 [ms]
Python+リスト	103.567
Python+NumPy	2.001

「スクリプト言語の簡便さ」と「高級言語の処理速度の速さ」の良いところ取りをすることができるため、Pythonはデータ解析や機械学習などの科学技術計算でよく利用されています。

処理性能

簡単な画像処理である2値化処理を例に、NumPyの性能を見てみます。

画像処理では、多くのデータを処理するため高速演算が求められます。例えば、300×200ピクセルのグレー・スケール画像の場合、60000個の画素値を扱うことになります。画像の2値化処理では、しきい値により画像の明るさの値を255(真っ白)と0(真っ黒)の2値に変換します(図1)。

● 比較1…コード効率

画像の2値化を以下の3種類で記述しました。

- C言語 [リスト1(a)]
- Python + リスト [リスト1(b)]
- Python + NumPy の配列 [リスト1(c)]

Pythonには、C言語の配列と似たような機能として「リスト」が標準で備わっています。Pythonのリストは「要素数が可変」で「要素ごとのデータ型が異なってもよい」ため、使い勝手が良い反面、処理速度が遅いという欠点があります。

画像処理や機械学習など、大量のデータを処理する