

地球の裏側からもOK! 360°見守りローバー君だぜ

ラズパイ式走る リモート探査カメラ

第3回 リアルタイム画像処理のための高速化テク②
…スループット/遅延性能のチューニング



森岡 澄夫

前回紹介

お品書き

1. OpenCVによる顔検出速度をラズパイ①と②とで比較
ここからラズパイ2
2. マルチコア向けに並列化されたOpenCV顔検出を試す
3. カメラ画像取得や顔検出処理などを複数のスレッドで実行する
4. パイプライン化によるスループット向上を試す
5. 4のパイプライン化+並列化OpenCVライブラリを試す
6. 画面分割によるレイテンシ短縮を複数スレッドで実現

今回紹介

図1 リアルタイム画像処理のためにラズベリー・パイ2のマルチコアを利用して試す並列化処理あれこれ

マルチコアであるラズベリー・パイ2を使って画像処理の高速化について検討します。ラズベリー・パイ・クラスのCPUにとって画像処理はかなり荷が重いので、マルチコアによる高速化は重要な検討課題です(図1)。

前回(第2回, 2015年12月号)は複数スレッドの効果や並列化OpenCVライブラリの効果を確認しました。ここまでは並列化すればそれで処理が速くなる、という趣旨で説明してきました。しかし何をどう速くする必要はあるかについて、もう少し細かく考えて

みましょう。次の二つをよく検討する必要があります。

高速化するための基本検討項目

■最初に検討すべき2項目

●その1…スループットとレイテンシのどちらを改善したいのか

画像処理の場合、スループットはフレーム・レートに、レイテンシはある1枚のフレームの処理にかかる遅延時間に対応します。例えば「お宝をカメラで監視して、取られそうになったら捕捉メカで犯人を捕まえる」アプリケーションを作るとします(図2)。

フレーム・レートが高ければ、見つからないよう素早く移動した犯人でも画像に映ります。また、レイテンシが短ければ(=反応までの遅延が短い)、犯人がお宝をつかんだ瞬間にメカを駆動できます。しかし、スループットが30フレーム/秒などと高くても、レイテンシが10秒もあっては、画像認識が完了した時点で犯人がすでに立ち去っており、捕り逃してしまいます。ここから分かるように、両者はあくまで別の性質の話です(8)。

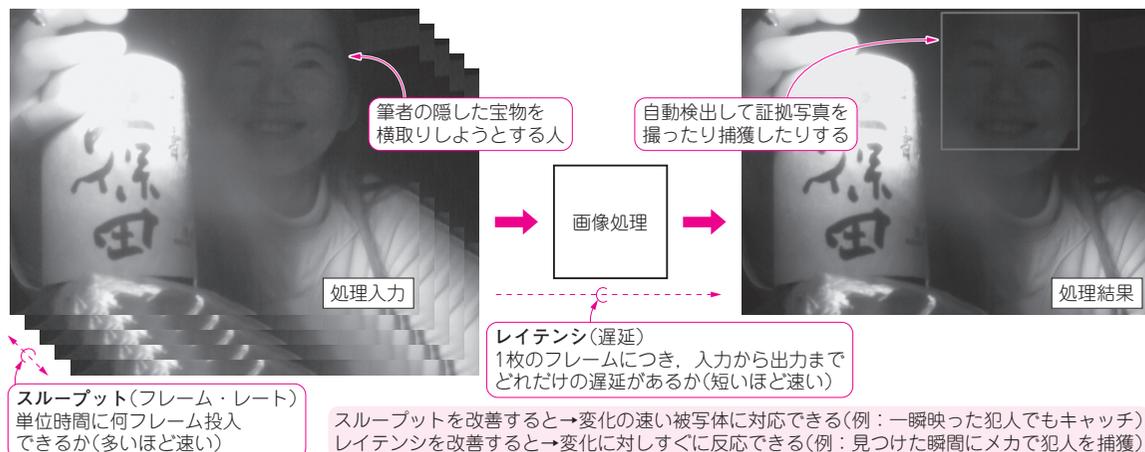


図2 スループットとレイテンシのどちらを向上させるか方針を決めないといけない
フレーム・レートが高くレイテンシも少ないことが望ましいが処理能力が求められる