

# ファットクライアントとクラウドを用いた カメラ画像データ解析フレームワークの提案

黒崎 裕子<sup>†</sup> 竹房 あつ子<sup>‡</sup> 中田 秀基<sup>‡</sup> 小口 正人<sup>†</sup>

<sup>†</sup>お茶の水女子大学 <sup>‡</sup>産業技術総合研究所

## 1. はじめに

近年カメラやセンサが手軽に利用できるようになり、一般家庭でライフログを取得して、防犯対策やお年寄りや子供のための安全サービスを目的としたライフログ解析アプリケーションが数多く開発されている。それらのアプリケーションを一般家庭で採用する場合は、サーバやストレージを設置して解析までを行うことは難しいため、センサ側計算機で全て処理せず、クラウド側での処理が必要となることが考えられる。我々の既発表研究 [1] では、画像の特徴抽出等の前処理部分から解析まですべてクラウド上で行っていたため、データ取得から解析結果の出力に時間がかかっていた。

本研究では、クライアント側にある程度の機能をもたせるファットクライアントモデルを適用し、センサ側で解析に必要な前処理を行って特徴量のみをクラウドに送信して解析することで、処理時間の高速化を目指す。本研究で実装したアプリケーションでは、前処理として撮影した画像を画像処理ライブラリ OpenCV[2] を用いて特徴点抽出を行い、そのデータをもとに Bag-of-Features 手法 [3] を用いてベクトルデータにする。そのベクトルデータを入力値として Jubatus[4] に学習させ、人の行動の判別する。既発表研究 [5] では、計算機の性能がセンサ側とクラウド側で同じ場合、クラウド側で前処理を行うと、データ転送コスト分だけ実行時間が増加した。センサ側計算機の性能が低い場合は、端末性能の分だけ前処理の速度が低下してしまうため、実行時間は大幅に増加する結果となった。よって、本研究では遅延が発生する場合や複数センサを用いた場合も考慮してを行い、センサ側での前処理が有効であることを示す。

## 2. ライフログ解析アプリケーションの設計

本節では本研究で実装したライフログ解析アプリケーションの概要を説明する。実装したアプリケーションは大きくわけて次の3つのプロセスに分かれている。

1. Motion を用いたカメラからの画像取得
2. Bag-of-Features を用いた特徴抽出
3. Jubatus での解析

Proposal of camera image data analysis framework using fat client and cloud

Yuko Kurosaki<sup>†</sup>, Atsuko Takefusa<sup>‡</sup>, Nakada Hidemoto<sup>‡</sup> and Masato Oguchi<sup>†</sup>

<sup>†</sup>Ochanomizu University <sup>‡</sup>National Institute of Advanced Industrial Science and Technology

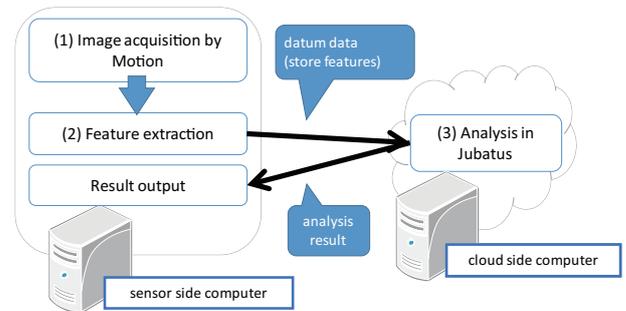


図 1: 実装概要 (センサ側での特徴抽出)

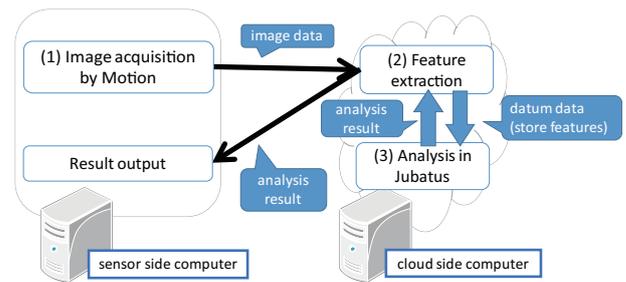


図 2: 実装概要 (クラウド側での特徴抽出)

(1) では、Web カメラサーバソフト Motion が動体検知すると取得した画像を随時保存し、画像を保存した際に指定したスクリプトファイルを実行するように設定した。(2) では、Jubatus での解析の前処理として、(1) で取得した画像を OpenCV[2] を用いて特徴抽出し、Bag-of-Features による画像データのベクトル化を行い、ベクトル化したデータを (3) Jubatus での解析プロセスに転送する。(3) では、(2) から受け取ったベクトルデータをもとに classifierAPI を用いて解析を行う。

センサ側の実装は、図 1 に示すように、(1), (2) を、センサ側で、(3) をクラウド側で行う。一方、クラウド側の実装では、(1) をセンサ側で、(2), (3) をクラウド側で行う。

## 3. 評価実験

(2) の前処理を、センサ側計算機、またはクラウド側計算機で実行し、その処理速度を評価する。実験では、実装したアプリケーションを用いて「ドアを開けた」「イスに座った」の2種類の人の行動を判別する。今回の評価実験では、実際にライフログ解析アプリケーションが使われる状況として、遅延が発生する場合や、複数センサからクラ

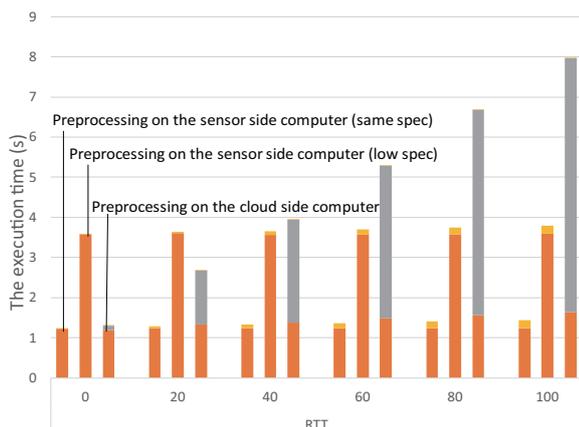


図 3: 遅延を考慮した評価実験結果

クラウドにデータが集中する場合を想定して、遅延を考慮した評価実験と、2台のセンサを用いた場合の評価実験を行う。

クラウド側計算機は、Intel Xeon CPU 3.10GHz を使用し、OS には Linux 2.6.32-5-amd64 Debian GNU/Linux 6.0.4 を用いた。センサデータ取得マシンには2種類のスペックのものを用意した。1つ目のマシンはクラウド側計算機を同スペックのもので、2つ目のマシンは Intel Core 2 Duo E6400 2.13GHz を使用し、OS には Ubuntu 12.04 LTS を用いた。また、Jubatus v. 0.5.1, OpenCV v. 2.4.7 を使用した。動画取得には WEB カメラの UCAM-DLV300T[6] を使用し、画像サイズは 640 × 480 とした。特徴抽出における Visual Words 数は 100 に設定し、辞書作りを行った。実験では、予め 100 枚の画像を用いて「ドアを開けた」状態と「イスに座った」状態を学習させた後、カメラから得られた画像データから特徴抽出を行い、生成されたベクトル値を datum に格納して Jubatus に転送した。

### 3.1 遅延を考慮した評価実験

クラウド側計算機とルータ間に dummynet を設置し、RTT を 0~100ms に設定して実験を行った。図 3 が実験結果である。遅延が大きくなるとデータサイズの大きい画像データの転送に時間がかかるため、クラウド側計算機での前処理を行う場合では、遅延が大きくなるにつれて解析時間も増加した。一方、画像データに比べてデータサイズの小さい特徴ベクトルを転送するセンサ側計算機での前処理では、解析時間に大きな変化はなかった。

### 3.2 2台のセンサを用いた評価実験

2台のセンサを設置してライフログの解析を行う評価実験を行った。センサ側計算機で前処理を行う場合、2台のセンサ側計算機を用意し、それぞれからセンサデータを取得し、前処理を行ってクラウド側計算機に特徴ベクトルを転送する。クラウド側計算機で前処理を行う場合は2台のセンサ側計算機から転送された画像データを、クラウド側計算機上で同時に前処理を行う。センサ側計算機を2台ともクラウド側計算機のスペックに統一した。結果は図4のようになる。クラウド側計算機での前処理は、転送された

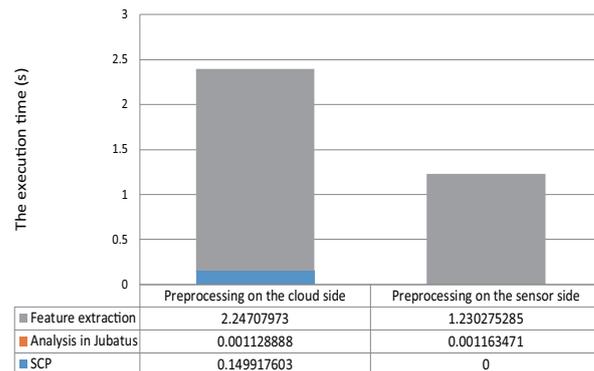


図 4: 2台のセンサを用いた評価実験結果

画像データの前処理をクラウド側計算機で同時に行うので処理速度が低下することが分かった。

## 4. まとめと今後の予定

本研究ではファットクライアントとクラウドを用いたライフログ解析の高速化を目的として、オンライン解析可能なライフログ解析アプリケーションの実装、評価実験を行った。実験結果より、高遅延環境やセンサ数が多い場合、すべてのデータをクラウド側計算機に転送して処理を行うのは、不利であることが示された。また、実行時間の多くは特徴量抽出等の前処理部分であることから、センサ側計算機がある程度の性能をもつ場合は、センサ側計算機にて前処理を行ってからクラウド側計算機へ転送する方が有利であることがわかる。よって、今後はセンサ側計算機での前処理部分の処理時間を低下させる手法の検討を行いたい。具体的には、Apache Storm 等のストリーム処理フレームワークを利用してアプリケーションを実装する。そして、既存アプリケーションとの評価実験も行いたい。

## 参考文献

- [1] 黒崎裕子, 山下暁香, 小口正人:”オンライン機械学習フレームワーク Jubatus によるライフログからの情報抽出”, DEIM 2014, D3-4, 2014年3月.
- [2] OpenCV, <http://opencv.org/>.
- [3] T. Nagahashi, H. Fujiyoshi, “Object Category Recognition by Bag-of-Features using Co-occurrence Representation by Foreground and Background Information”, Machine Vision Applications, pp.413, 2011.
- [4] Jubatus, <http://jubat.us/ja/>.
- [5] 黒崎裕子, 竹房あつ子, 中田秀基, 小口正人:”クラウド上でのライフログ解析のためのオンラインフレームワーク Jubatus を用いたアプリケーション実装”, DICOMO 2014, pp.254-257, 2014年7月.
- [6] UCAM-DLV300T, <http://www2.elecom.co.jp/multimedia/pc-camera/ucam-dlv300t/>.