

スマートフォンでは、1)コンテキスト推定のためのセンシング、2)推定されたコンテキストに基づくデスクトップ制御、を行う。

センシング部は、スマートフォンに内蔵された各種のセンサ(GPS、加速度、ジャイロ、NFC など)を、必要に応じて利用し、センシング結果をサーバへアップロードする。

デスクトップ制御部は、サーバから通知された内容に基づき、スマートフォンのデスクトップに必要なアプリのみを表示する。スマートフォン側にも、サーバと同様のコンテキスト毎のアプリリストを保持し、通知内容と照合して、必要に応じて内容の修正を行う。また、アプリリスト内のうち、所定の待ち時間経過後に、特定のアプリを自動起動することも可能とする。

3. システム試作

上記のコンセプトに基づき、システムの試作を行った。

3.1 サーバ

本試作では、コンテキスト推定-アプリ管理部(以下、アプリ管理部)と、通知部を別サーバとする構成を取り、HTTPで通信することとした。

アプリ管理部は、コンテキスト推定とアプリ管理を一体化し、スマートフォンから送られる NFC(Near Field Communication)-ID 値に基づき、コンテキストとアプリリストを決定する簡単な仕様とした。スマートフォンからアプリ管理部への送信も、HTTPでの通信とした。アプリ管理部は、javaによる実装とした。

通知部は、アプリ管理部と HTTPで通信して、必要な情報を受け取り、スマートフォンにプッシュ通知する。通知部は、javaとCで実装した。

3.2 スマートフォン

スマートフォンには、プログラム開発の自由度の高い Android 2.1 搭載スマートフォンを用いた。

デスクトップ制御部は、Androidにおけるホームアプリとして作成した。通知受信部から、コンテキストやアプリリストを intent で受信して、コンテキストに応じたアプリのみが表示されるようデスクトップを変更する。アプリ自体が通知された場合には、デスクトップにアイコンの追加も行う。図2に変更時の例を示す。

コンテキストのセンシングには、外付けの NFC リーダライタを Bluetooth で接続して利用した。NFC は、“かざす”という分かりやすい操作で情報取得が可能である。入退室管理などでの普及が見込まれ、コンテキストセンシングの重要な要素になると考えている。本試作では、



図2 デスクトップの変化

200msec 周期で読み取りを行い、NFC タグが発見された場合に、そのタグ ID を通知することとした。

3.3 評価

試作したシステムを用いて、(1)NFC 検知からスマートフォンに通知が行われるまでの時間、(2)デスクトップでのアプリ選択時間、を計測した。実験結果を表1に示す。

項目(1)は、サーバとスマートフォンとで処理を分けたため、ユーザに対するシステムのレスポンスを見積るために行った。実験の結果、NFC リーダライタを取り付けたスマートフォンを NFC タグにかざして、サーバから通知が行われるまでの時間は、平均 1.6 秒(95%信頼区間 1.4~1.8 秒)であった。即時性は十分でないが、実運用時には、数十秒以上の大きな遅延がなければ問題にならないと考えている。

項目(2)は、ユーザがアプリを選択するまでの時間が、デスクトップに並ぶアプリのアイコン数と関連するかを確認するために行った。スマートフォンでは、アプリは画面全体に多数配置されることが多い。本試作では、コンテキストによりデスクトップ上のアプリアイコンを必要最低限にすることで、ユーザの操作性向上を狙っている。

そこで、画面一杯にアイコンを表示した場合(アイコン数 24 個)と一行分 4 個の場合で選択時間を計測した。

実験の結果、アイコン数 24 個の場合、平均 4.1 秒(95%信頼区間 3.4~4.8 秒)であり、アイコン数 4 個の場合、平均 2.1 秒(95%信頼区間 2.0~2.2 秒)であった。平均値で 2 秒差があり、アイコン数が増えるとはらつきも大きくなることが分かった。コンテキストによりアプリ数の絞り込みができれば、操作性の向上につながることを確認できた。

表1 実験結果(平均値)

NFC 検知からの通知時間	1.6 秒 (95%CI 1.4-1.8 秒)
選択時間-アイコン数 24 個	4.1 秒 (95%CI 3.4-3.8 秒)
選択時間-アイコン数 4 個	2.1 秒 (95%CI 2.0-2.2 秒)

4. まとめ

ユーザのコンテキストに応じて、最適なアプリ・データを適切なタイミングでスマートフォンに提示する、“コンテキストデスクトップ”システムを検討、試作した。

コンテキスト処理をサーバで実施し、スマートフォンに通知する構成で、レスポンスが 1.6 秒と実用上問題にならないことが分かった。また、アプリの絞り込みができれば操作性の向上につながることも分かった。

今後、コンテキストセンシングの追加やコンテキスト推定の高度化を検討し、システムの実用化を図る。

参考文献

- [1]上坂 大輔, 村松 刺激, 横山 浩之, “携帯電話におけるユーザ状況適応型操作支援技術の検討”, 情報処理学会 研究報告, 2008-UBI-20(6)
- [2] P.Coppola, V. Della Mea, L. Di Gaspero et al., “The Context-Aware Browser”, Intelligent Systems, IEEE, Jan.-Feb. 2010, Vol-25, p.38-47