

HTML5 APIにより取得可能なデバイス情報を利用した端末識別手法の提案と実装

高須 航† 磯 侑斗‡ 桐生 直輝‡ 齋藤 孝道†
明治大学† 明治大学大学院‡

1. はじめに

行動ターゲティング型広告のような、利用者のインターネット上での行動を追跡することで利用者の興味関心を推測し、情報を提供するサービスが増えてきている。利用者を一意に識別することにより、サービス提供者は利用者の行動を追跡することを実現する。

Web 通信では一般に、利用者を一意に識別する手法として、HTTP クッキーを用いる手法がある。しかし、Web ブラウザは HTTP クッキーの削除や拒否をする機能を備えており、追跡を回避することが可能である。

一方で、HTTP クッキーを用いることなく、利用者の Web ブラウザの特徴や HTTP 通信時の特徴を用いることで利用者を推定する手法が、Peter Eckersley[1]によって提示されている。この論文では、94.2%の精度で端末を一意に識別可能であることが示されている。しかし、そこで提示された手法においては、Web ブラウザの特徴を利用することから、利用者が Web ブラウザを変更してアクセスした場合、追跡が困難となる。

そこで本論文では、Web ブラウザに依存しない新たな特徴点として JavaScript から採取できるデバイス情報を用い、利用者が Web ブラウザを変更しても同一端末として識別可能な特徴点を提案し、その実装を示す。

なお、本論文で言うデバイスとは、CPU や HDD などコンピュータを構成する各装置や、ディスプレイ、カメラのような端末に付随する IO 機器を指す。

2. Web 上での行動追跡

Web 上で利用される、二つの行動追跡手法について説明する。なお本論文では、Web ブラウザから採取することが可能で、かつ、端末の特定に繋がるような情報を特徴点と呼ぶ。

A Method of Estimating a Device by Hardware Characteristic Obtained with HTML5 API

†Ko Takasu

‡Yuto Iso ‡Naoki Kiryu

†Takamichi Saito

Meiji University(†), Graduate school of Meiji University(‡)

2.1 HTTP クッキーによる Web トラッキング

ある広告画像が埋め込まれた Web サイトにアクセスすると、その広告画像を提供する広告サーバから HTTP クッキーが発行され Web ブラウザに保存される。以降、同じ広告サーバが提供する広告画像が埋め込まれた Web サイトにアクセスすると、Web ブラウザは保存していた HTTP クッキーを広告サーバに送信する。HTTP クッキーと利用者がアクセスした Web サイトの URL を紐付けることで、広告サーバは利用者の Web 上での行動を追跡することが可能となる。このような行動追跡手法を、本論文では HTTP クッキーによるトラッキングと呼ぶ。

利用者毎に一意な識別子を割り当てるので、確実に利用者を特定できるが、利用者が HTTP クッキーを削除、または拒否した場合追跡が不可能となる。

2.2 Fingerprint によるトラッキング

Web 通信において、User-Agent や Accept-Language のような Web ブラウザのリクエストヘッダに含まれる特徴点や、Web ブラウザのプラグインや、画面解像度といった情報を、利用者の Web ブラウザや端末の特徴点とすることができる。このような特徴点の集合を本論文では Fingerprint と呼ぶ。これは Web ブラウザ上で JavaScript や Flash などを実行させることで採取可能である。採取した Fingerprint と利用者がアクセスした Web サイトの URL を紐付けることで、トラッキングが可能となる。このような Fingerprint を収集し行動追跡を行う手法を、本論文では Fingerprint によるトラッキングと呼び、デバイスの Fingerprint の採取法を提案する。

これまで、デバイスの Fingerprint として、文献[2]では、CPU コア数、GPU レンダリングの有無、メディアデバイスの有無、タッチディスプレイか否かの情報を採取しており、文献[3]では、CPU 拡張命令対応の有無の識別に成功している。

HTTP クッキーなどの識別子を用いないため、利用者は Fingerprint によるトラッキングを防ぐことは難しいが、特徴点の類似した端末が存在する場合、一意に識別することが出来ないとされている[5]。

3. デバイスの Fingerprint による端末識別

3.1 概要

Web ブラウザから採取可能なデバイス情報を用いることで端末識別を行う。デバイス情報は利用者の端末毎に特徴が変化するので、利用者が Web ブラウザを変更してアクセスした場合でも、同じ端末として識別することが可能となる。JavaScript 及び JavaScript から利用可能な HTML5 API を用い、デバイス情報を採取する。

3.2 HTML5 API

HTML5 API という言葉にその明確な定義はなく、W3C や WHATWG の他、IETF や Khronos などにより策定された API も含めて HTML5 API と呼称されることもある。

本論文では、HTML5 のドラフト発表以降に策定された W3C 及び WHATWG の関与する API を HTML5 API と定義する。Javascript 及び HTML5 を用いた、特徴点の採取方法を提案する。

4. 提案手法

4.1 概要

端末を識別するためのデバイス情報として、ストレージ空き容量、ディスプレイのリフレッシュレート、バッテリー状況や、デバイスピクセル比を新たな特徴点として提案する。

4.2 提案する特徴点の採取方法

4.2.1 ストレージ空き容量

Web ブラウザのオリジン[4]毎に保存可能なデータ容量の上限値をバイト単位で任意に要求することが可能である、Quota Management API を用いる。この API で定義されている navigator.webkitTemporaryStorage.requestQuota メソッドによりデータ容量の上限値を要求すると、要求した上限値分の保存領域が確保できるか否かを確認することができる。確保できる場合、要求した上限値が返る。確保できない場合エラーメッセージが返るが、ストレージ空き容量を超えるような大きな値を要求した場合、確保可能な最大の上限値を要求した場合の上限値が返る。この上限値は、ストレージ空き容量を約 100MB の誤差で推測可能な値である。従って、誤差約 100MB の精度でストレージ空き容量を採取することが可能となる。

4.2.2 ディスプレイのリフレッシュレート

Animation Timing API を用いる。リフレッシュレートとは、ディスプレイが単位時間あたりに描画される回数を示す。この API で定義されている requestAnimationFrame メソッドは、ディスプレイが描画されるタイミングで実行される。現在の時刻を取得する Date.now() メソッドを利用し、requestAnimationFrame メソッドが実行された後の時刻と、次にそのメソッドが実行された後の時刻を取得する。それらの差が、ディスプレイの描画時間となる。ディスプレイの描画時間は、CPU

の負荷によって誤差が生じるので、この時間を繰り返し測定し、その平均値を計算することで、実際のディスプレイのリフレッシュレートから誤差約 $\pm 2\text{hz}$ の精度で推測することが可能となる。

4.2.3 バッテリー状況

Battery Status API を用いる。この API で定義されている、window.navigator.battery プロパティにおける、charging、chargingTime、dischargingTime、level メソッドを用いることで、バッテリーが充電中か否か、バッテリーの充電完了までの残り時間、バッテリーが利用できる残り時間、バッテリー残量を採取することが可能である。

4.2.4 デバイスピクセル比

デバイスピクセル比とは、画像の 1 ピクセルを端末のディスプレイ上で何ピクセルとしてレンダリングを行うかを表した比率である。window.devicePixelRatio プロパティにより、端末のディスプレイが何倍の比率でレンダリングを行うのかを数値として採取可能である。

4.3 提案する特徴点の考察

デバイスピクセル比は端末によって値が固定されているが、その他の特徴点は採取する度に値が変わる可能性がある。しかしディスプレイのリフレッシュレートは頻繁に変更するものではないためほぼ一定であること、また、ストレージ空き容量とバッテリー状況は、値の変動する許容範囲を決めることで、端末識別に用いることが可能と考えられる。

5. まとめ

端末を識別可能な新たな特徴点を提案した。提案した特徴点と既存の特徴点を用いることで、利用者が Web ブラウザを変更しても同一端末として識別することが可能であると考えられる。識別精度を計測する実験は今後の課題としたい。

6. 参考文献

- [1] Peter Eckersley, 2010, How Unique Is Your Web Browser?
<https://panoptlick.eff.org/browser-uniqueness.pdf>
- [2] 後藤浩行, 齋藤孝道, Web 行動追跡のためのハードウェア特徴点の抽出, 2013 暗号と情報セキュリティシンポジウム p72-77
- [3] 桐生直輝, 後藤浩行, 齋藤孝道, CPU 拡張命令対応の有無による CPU アーキテクチャの推測, 第 75 回情報処理学会全国大会講演論文集 3-613, 3-614
- [4] RFC6454 The Web Origin Concept
<http://www.ietf.org/rfc/rfc6454.txt>
- [5] 齋藤孝道, 磯侑斗, 桐生直輝, Web Browser Fingerprinting に関する技術的観点での一考察, 2014 年暗号と情報セキュリティシンポジウム予稿集.