明治大学総合数理学部

2019 年度

卒　業　研　究

偽造 Wi-Fi アクセスポイントによる現在地情報のスプーフィング攻撃の脅威

学位請求者　先端メディアサイエンス学科

江藤一樹

目次

第 1 章

はじめに

近年，スマートフォンの普及に伴い，ゲームや食事サイトなどのインターネットサービスで，デバイスを通して取得される位置情報の利用が増え，社会インフラとして必要性が高まった．その一方，GPS（グローバル・ポジショニング・システム）への悪意ある攻撃として，スプーフィング攻撃（なりすまし攻撃）の脅威が指摘されている [1]．これを受けた GPS は実際にいる位置とは異なる位置を示し，船の自動運転やドローンなどの社会インフラの混乱を引き起こす原因となる．2018 年 11 月 12 日に，店舗に訪れたかのように位置情報を偽り，イオンアプリの来店ポイントを不正取得したとして，男が逮捕された [2]．

デバイスは，周囲にあるアクセスポイント（AP）の MAC アドレスをサーバに送信し，AP の情報が格納されているデータベースと照合して，位置情報を推定する．従って，偽装された偽の MAC アドレスを与えられるとデバイスの現在位置を操作されてしまう恐れがあると考えられる．

そこで，本稿ではこの位置情報のスプーフィング攻撃の実現可能性を調査し，攻撃のリスクと攻撃を受ける条件を明らかにする．攻撃の検証の為に，Ubuntu 18.04 LTS が搭載された PC を 5 台用意し，ネットワークインターフェイスの MAC アドレスを偽装する．

本研究の新規性は次の通りである．

1. デバイスに対する位置情報のスプーフィング攻撃のリスクを明らかにすること．
2. 偽 AP の信号強度などの条件を明らかにする．
3. 位置情報スプーフィング攻撃への対策．

第 2 章

提案手法

本研究で検証する位置スプーフィング攻撃は，攻撃対象のデバイス付近に，異なる場所にある AP の MAC アドレスを偽装した偽の AP を複数設置することにより，現在位置を誤推定させることで実現する．実験システムの概要図を図 A.11 に示す．偽の ARP パケット a_3，a_4，a_5 を受信したホストは，本来の a_1，a_2 よりも a_3 付近の位置と誤認する．偽の AP には Ubuntu 18.04 LTS，MAC アドレスの変更に macchanger[4] を使用する．

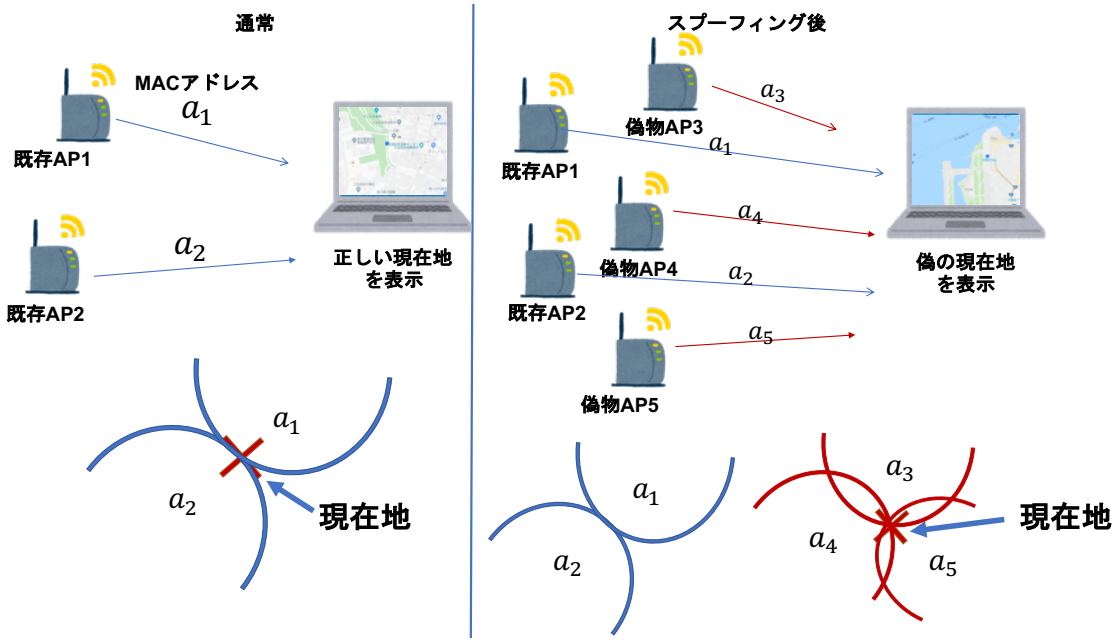

図 2.1　提案手法の概要図

2.1 Geolocation API[3]

W3C（World Wide Web Consortium）によって標準化された，Web からデバイスの位置情報を取得する API である．API の概要図を図 3.1 に示す．JavaScript の navigator.geolocation オブジェクトを通じて提供される．デバイスがこのオブジェクトに対してメソッドを実行する時に，実行したデバイスの IP アドレス，周囲の Wi-Fi の MAC アドレス，GPS の情報などを API を介して，サーバーにリクエストを送り，位置情報を取得する．Geolocation API を使用するには，通信が https であることが必須であるため，サーバに送られている内容を盗聴することを困難にしている．

図 2.3 は Geolocation API を使用したスクリプト，図 2.4 はそのスクリプトの結果である．1 行のスクリプトで緯度と経度，精度等が取得できる．

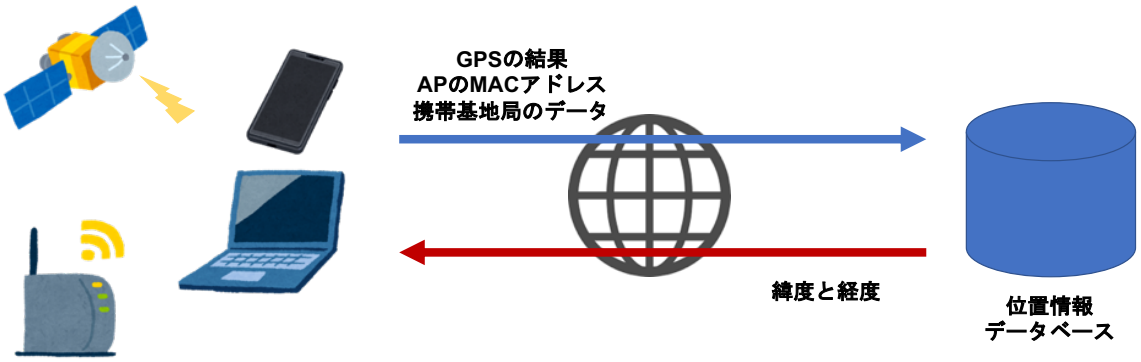

図 2.2　Geolocatoin API の概要図

```
1  <!DOCTYPE html>
2  <html lang="en" dir="ltr">
3    <head>
4      <meta charset="utf-8">
5      <title></title>
6    </head>
7    <body>
8      <script type="text/javascript">
9        navigator.geolocation.getCurrentPosition(function(e){
10         console.log(e);
11       })
12     </script>
13   </body>
14 </html>
15
```

図 2.3　Geolocatoin API のスクリプト例

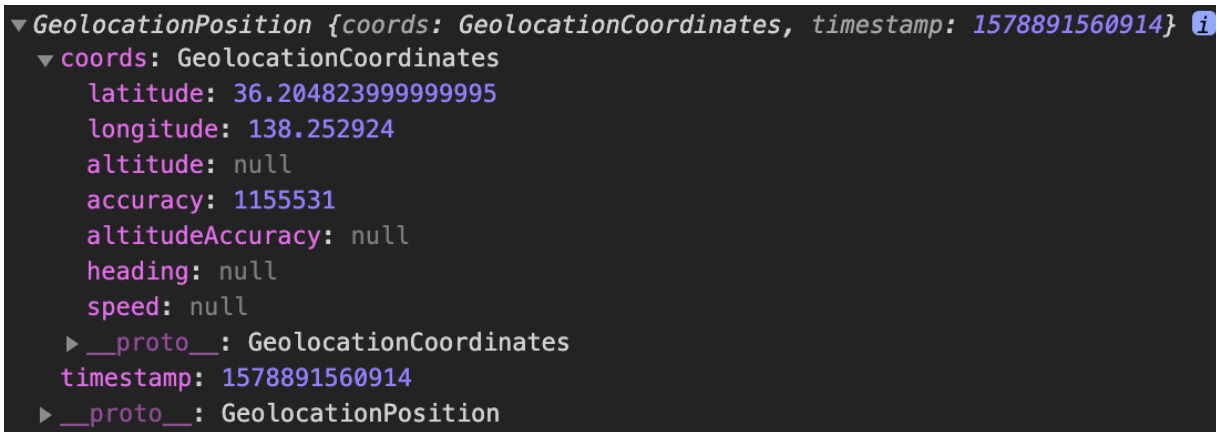

図 2.4　Geolocatoin API の結果

2.1.1　位置情報の推定に利用される要素の調査

Geolocation API が実行された時，IP アドレスと AP が位置推定に利用されているのか調査を行う．調査方法は，無線 LAN アダプターを搭載していないデスクトップ PC を対象に，グローバル IP が変化した時と無線 LAN 子機を付けた時で位置情報が変化するかの 2 つである．グローバル IP の変更はルータの再起動によって行われる．

2.1.2　調査結果

グローバル IP が変化した時の結果を表 2.1 に示す．この結果から，Geolocation API が位置推定にグローバル IP を利用していることが分かる．無線 LAN 子機を付けると正常な位置を指し，デバイス周辺の AP を利用していることが確認できる．

表 2.1　グローバル IP が変化した時の結果

日付	グローバル IP	位置情報の場所
11 月 2 日	126.99.236.27	秋葉原駅付近
11 月 3 日	126.51.149.135	曙足駅付近
11 月 4 日	126.99.238.57	足立区役所付近
11 月 5 日	126.99.227.208	渋谷駅付近
11 月 7 日	126.74.174.194	浅草橋駅付近

2.2　AP 情報の取得

AP の数と情報を取得する為に，macOS の “airport” コマンドを使用する．このコマンドは，デバイス周辺にある AP の SSID，MAC アドレス，信号強度，チャンネルを含む 7 つの情報を表示する．

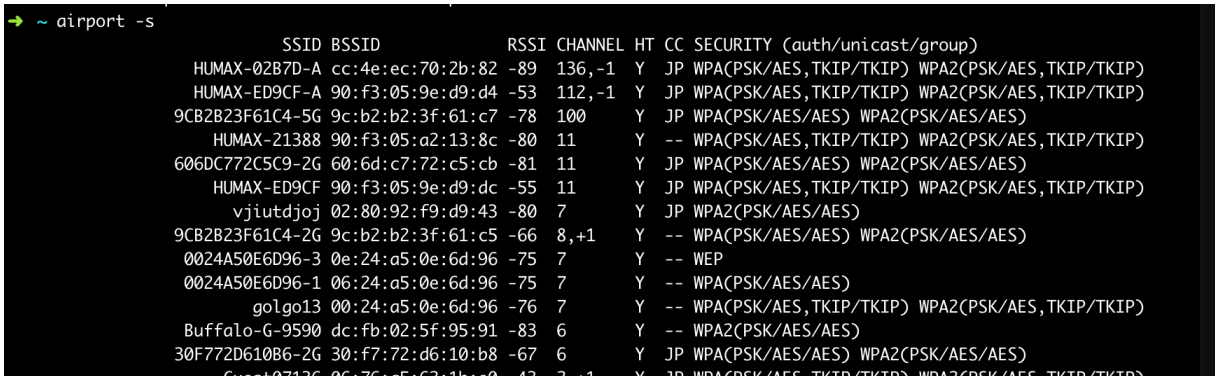

図 2.5　airport コマンドの実行結果

2.3　MAC アドレスの偽装

ネットワークインターフェイスの MAC アドレスを任意な値に変更する Linux のパッケージである macchanger[4] を使用する．図 2.6 は，macchanger によって偽装された偽 AP の MAC アドレス aa:aa:bb:bb:cc:cc を含む ARP パケットである．

図 2.6　MAC アドレスを偽装した ARP パケット

2.3.1　Windows の MAC アドレスの偽装

Windows10 のアクセスポイント化と MAC アドレスの変更が可能であるが，変更する MAC アドレスに条件がある [5]．MAC アドレス XY:XX:XX:XX:XX:XX の X は全ての 16 進数，Y は 2, 6, A, E でのいずれかでなくてはならない．図 2.7 は，全ての X に F，Y に 2, 6, A, E を入れ，偽装し，airport コマンドで取得した結果である．第一オクテットの F が 5 に変更されているのは興味深いが，2, 6, A, E の時は偽装され，それ以外の値では，元の MAC アドレスが取得された．

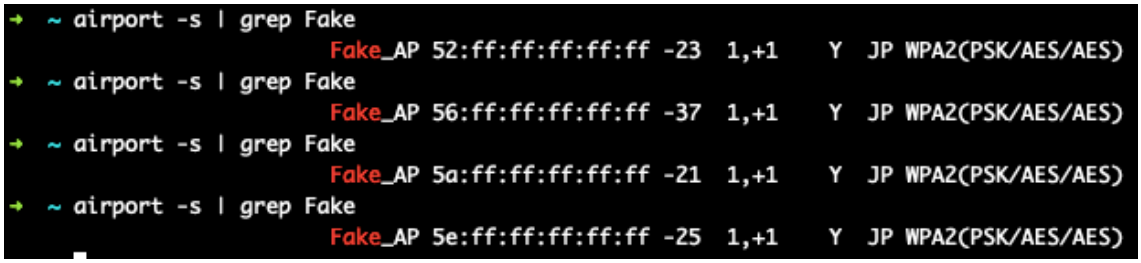

図 2.7　Windows による MAC アドレスの偽装結果

第 3 章

実験

位置情報の偽装が可能な条件を調べるために，以下の条件で実験を行う．

1. 実験 1 : PC とスマートフォンによる位置情報の観測とスプーフィング実験.
2. 実験 2 : 偽 AP と対象デバイスの距離，AP の数についてのスプーフィング実験.

3.1　実験環境

位置スプーフィング攻撃が成功するためには，既に存在する AP 数を上回る必要があると仮定し，偽 AP を 5 台用意する．偽 AP の MAC アドレスは，異なる場所にある周囲の AP から，攻撃者が信号強度の強い順に 5 台選択し，それぞれの MAC アドレスに偽装する．

3.2　偽装対象のインターネットサービス

本研究では，ユーザの現在地を地図上に表示する次の 2 つのサイトを利用する．Googole 社が提供している地図サービス “Google Maps”．すかいらーくグループの公式サイトが提供している「ガスト店舗検索」サービスである．

3.2.1　地図 API を使用したサイトの調査

地図 API を使用している主要サイトの中で Geolotion API を利用しているサイトの割合を調査した．305 件調べ，Geolocation API を利用有無の結果を表 3.1，地図 API の種類別の利用頻度を表 3.2 に示す，に示す．305 件中，Geolotion API を使用してるサイトは 18 件である．そのうちの 17 件はユーザの周辺にある店舗検索に利用されており，残りの 1 件がすかいらーくグループの公式サイトである．

表 3.1　Geolotion API の有無

サイト名	API	Geolotion API の有無
Hot pepper グルメ	Google Maps	無
マクドナルド	Google Maps	有
スターバックス	MAPPLE	有
すかいらーくグループ	Google Maps	有
⋮	⋮	⋮
Retty	Yahoo map API	無
計		18/305

表 3.2　地図 API 別の利用頻度

Google Map API	274
いつも NAVI	12
Yahoo map API	7
その他	12

3.3　実験方法

3.3.1　実験 1：PC とスマートフォンを用いた位置情報の観測

実験 1 では，GPS の有無，異なるインターネットサービスで偽装結果に変化があるのか調べる．そのために，GSP が搭載されているスマートフォンと搭載されていない PC で検証する．現在地を取得する際，デバイスをオンラインとオフラインの二つの状態にする．実験条件を表 3.3 に示す．

実験場所は，本研究室，自宅，江古田の森公園付近，本学地下室の四箇所である．偽 AP が偽装する MAC アドレスは，東京都小金井市のたけのこ公園付近で入手した AP の MAC アドレスである．偽装対象のデバイスに，PC は Mac book pro，スマートフォンは iPhone SE を使用する．

既存 AP と偽 AP の数の関係性を明らかにする為，実験場所の既存 AP 数を調べる．5 秒間隔で，PC を用いて AP の数を 100 秒間取得する．一つの AP がチャンネル毎に異なる MAC アドレスを所持している場合，その MAC アドレスの数を AP の数とする．

本実験では，他の場所の AP の MAC アドレスに変えても偽装が可能か調べる．偽装に使用する AP は，京都府にある法然寺付近と沖縄県の那覇空港，香港の三ヶ所である．本実験では攻撃対象を PC のみ，実験場所を本学地下室，偽 AP の数を 5 台とする．

表 3.3　実験条件

	機種	オフライン	オンライン
スマートフォン	iPhone SE（IOS 12.4.1）	機内モード	LTE or Wi-FI 接続あり
PC	Macbook Pro（macOS Catalina）	接続なし（Wi-Fi off）	デザリング or Wi-Fi 接続あり

3.3.2　実験 2：偽 AP と攻撃対象デバイスの距離，偽 AP の数による影響

環境を変えて調査を行い，位置情報の偽装ができるか調べる．偽 AP と攻撃対象の距離を 0m，3m，6m，9m，12m と変化させた時の信号強度と位置情報の偽装結果を調べる．信号強度は，50 秒間に 10 回取得し，その結果の平均とする．

偽 AP と攻撃対象の距離を 0m にし，偽 AP の数を 5 1 まで変化させて，位置情報が偽装される数を調べる．

3.4　実験結果

3.4.1　実験 1

実験場所の AP 数を表 3.4，位置スプーフィング結果を表 3.5 に示す．ここでの〇は偽装できたこと，×は正常の位置推定が行われたことを示す．

表 3.5 より，位置スプーフィング攻撃が成功したのは，本学地下室で PC を対象とした時だけで，実験条件での結果に違いはない．その為，位置情報の偽装に最も関係する条件は，既存 AP と偽 AP の数である．

スマートフォンに対する位置情報の偽装は，実験環境によらず，できないことが分かった．AP の情報ではなく，GPS か携帯基地局のセンサーに基づいて位置を推定していると考えられる．

位置スプーフィングされた偽装結果を図 3.1（すかいらーく）と図 3.2（Google Maps）に示す．両サイトで同じ場所を指したことから，Geolotion API を利用しているサービスに対して，位置を偽装できることが分かった．

表 3.4　実験場所の AP 数

場所	AP の数	
	μ	σ
研究室	123	11.8
自宅	63	4.8
公園	19	4.8
地下室	2	0.4

表 3.5　実験 1：観測結果

デバイスの状態	GPS	実験場所	Google Map	すかいらーく
オンライン	あり	研究室	×	×
		自宅	×	×
		公園	×	×
		本学地下室	×	×
	なし	研究室	×	×
		自宅	×	×
		公園	×	×
		本学地下室	○	○
オフライン	あり	研究室	×	×
		自宅	×	×
		公園	×	×
		本学地下室	×	×
	なし	研究室	×	×
		自宅	×	×
		公園	×	×
		本学地下室	○	○

表 3.6　実験 1：位置スプーフィング結果

AP 元の場所	Google Maps	すかいらーく
京都	○	○
沖縄	○	○
香港	○	○

3.4.2　実験 2

偽装対象のデバイスと偽 AP の距離を変えた位置スプーフィングの結果を表 3.7，偽 AP 数を変えた結果を表 3.8 に示す．ここでの○は位置情報を偽装できこと，×は正常の位置推定が行われたことを示す．両結果から位置情報の偽装に必要なのは，既存 AP 数より多い偽 AP 数を設置することだと分かる．

表 3.7　実験 2：距離を変えた実験結果

結果	距離 [m]	偽 AP										既存 AP			
		a[dBm]		b[dBm]		c[dBm]		d[dBm]		e[dBm]		a'[dBm]		b'[dBm]	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
○	0	-36.4	0.68	-39.8	2.6	-35.2	2.2	-32.3	4.3	-41.7	1.2	-89.2	1.2	-90	1.5
○	3.0	-55.6	4.1	-54.2	2.0	-58.7	2.7	-51.7	2.9	-56.3	1.8	-85.4	1.9	-85.6	1.4
○	6	-63.6	1.8	-56.1	2.6	-57.8	2.9	-56.6	1.6	-61.8	2.0	-86	3.2	-85.6	3.1
○	9	-66	1.4	-66.6	0.5	-64.4	1.8	-60.9	1.4	-59.3	1.4	-88.6	1.0	-88.1	2.6
○	12	-69.3	1.8	-61.1	2.4	-65.1	1.0	-57.4	1.7	-64.6	2.1	-87.4	1.9	-87.4	1.6

図 3.1　実験 2 の偽装結果（那覇）

図 3.2　実験 2 の偽装結果（香港）

表 3.8　実験 2：偽 AP 数を変えた実験結果

結果	偽 AP 数
○	5 台
○	4 台
○	3 台
○	2 台
×	1 台

3.5　考察

3.5.1　実験 1：観測

GPS ありのデバイスの位置情報は，実験環境によらず偽装されない．GPS の電波が届きにくい地下室の実験では図 3.3 の様に，位置情報の精度は悪くなるが，実験場所付近を指している．従って，位置情報の推定には，AP の結果より GPS の結果を優先していることが分かる．

オンラインに接続をしていない状態でも正常の位置を指した．これはブラウザにキャッシュされた情報から位置を推定していると考える．

図 3.3　スマートフォンを用いた地下室での観測結果

3.5.2　実験 1：PC による実験

表 3.6 より，位置情報の偽装ができる場所は，AP の MAC アドレスがデータベースに格納されていれば，どこでも可能だと考える．両サイトによる結果に違いがなかったのは，使用しているブラウザが同じな為，照合を行っているデータベースも同じだからと思われる．

この実験により，第三者の位置情報に対する偽装だけでなく，[2] のニュースのように自身のデバイスの位置情報を全国各地に偽装し，不正に利益を得ることができてしまうことが現実的であることが実証された．

3.5.3　実験 2

既存 AP の信号強度は非常に弱い電波であるが，既存 AP が偽 AP の数を上回った時に，実験場所付近を指したことから，位置スプーフィングに信号強度が関係しないと考える．その為，電波を拾えれば 12m より距離を離しても偽装ができてしまうことが分かる．

第 4 章

対策

本リスクへの対策は以下である．

1. GPS の使用を必須にする
2. 多様なリソースからの推定
3. キャッシュ，履歴からの推定

本研究で，GPS が搭載されているデバイスに対する偽装ができなかったことが示された．従って，ドローンや自動運転の乗り物などの位置情報を利用したデバイスに対し，GPS を搭載することが効果的である．しかし，GPS の信号が受信できない環境から安全に位置推定をするために，Wi-FI などの位置情報に使えるリソースを複数組み合わせる必要がある．スプーフィング攻撃により位置情報が急に変化した時，変化前のキャッシュや履歴から現在地を推定するか，ユーザに対して「位置情報が偽装されている可能性がある」等の警告を送るべきと考える．

第 5 章

おわりに

本研究では，インターネットサービスで利用される現在地を取得する機能に対して，位置情報のスプーフィング攻撃の実現可能性を調査し，攻撃のリスクと攻撃を受ける条件を明らかにした．位置情報の取得には，デバイス周辺にある AP の MAC アドレスが利用されていることから，既存 AP の数を上回る MAC アドレスを偽った AP を設置することで，GPS を搭載していないデバイスの位置情報を偽装できることが分かった．香港や那覇への偽装ができたことから，AP の MAC アドレスがデータベースに格納されていれば，偽装できる場所に制限なく偽装できる．このリスクに対する対策として，GPS の使用を必須にする，多様なリソースからの位置推定，キャッシュ，履歴からの推定の三つが考えられる．

今後の課題は，以下の三つである．

1. 異なるブラウザによる観測
2. 既存 AP と偽 AP を同数にした実験
3. サーバに送る内容を明らかにする

Google や Apple などのブラウザを提供している企業は，それぞれで AP の MAC アドレスと位置情報を紐づけたデータベースを構築している．その為，Geolotion API を実行した際，各ブラウザはそれぞれの企業が所持しているデータベースと照合していると予想する．そこで，照合を行うデータベースが異なるか調査し，位置スプーフィング攻撃のリスクを受けるデータベースを明らかにする．

実験 2 で，偽 AP が既存 AP の数を上回っている場合，位置情報の偽装ができると分かったが，偽 AP と既存 AP を同数にした場合，位置情報の偽装に信号強度が関係するのか明らかになっていない為，調査する必要がある．

謝辞

本研究を行うにあたり，多くの方より御指導いただきました．特に，多大なる御指導を受け賜りました，指導教官である明治大学総合数理学部先端メディアサイエンス学科の菊池浩明教授に深く感謝申し上げます．また，研究に使用するデータセットを提供して下さった菊池研究室の方々に深く感謝の意を表するとともに，謝辞とさせていただきます．

参考文献

[1] 海老沼，”GPS 信号の脆弱性と今そこにある危機”，中部大学工学部紀要，2017.

[2] 朝日新聞デジタル，“イオンから来店ポイント詐取容疑　ＰＣで位置情報偽装”，2018 年 11 月 12 日.

[3] Geolocation API Specification 2nd Edition（https://www.w3.org/TR/geolocation-API/，2019 年 12 月参照）

[4] ubuntu manuals（http://manpages.ubuntu.com/ manpages/bionic/man1/macchanger.1.html，2019 年 12 月参照）

[5] Change MAC address in Windows 7 or later for wireless adapter - Change MAC address - LizardSystems（https://lizardsystems.com/change-mac-address/articles/change-mac-address-windows-7-or-later-wireless-adapter/，2019 年 1 月参照）

[6] Google の位置情報サービスに登録されたアクセス ポイントを管理する（https://support.google.com/maps/answer/ 1725632?hl=ja　2019 年 12 月参照）

付録 A

顔検出防止技術の評価実験

A.1　はじめに

近年，店舗における顧客の購買行動推定や空港やイベント会場における犯罪防止などの目的に顔認証システムが広く普及してきている．その一方で，無条件にカメラで撮影されることや，取得された個人情報の用途が不明な事で生活者に生じる不安や属性推定といったプライバシーに関する課題も懸念されている．これに対して，山田ら [1] は，反射性のある素材により顔検出を防止するメガネ型デバイスのプライバシーバイザー（以下，バイザー）を提案している．

しかしながら，顔認証は本人の協力の下で行う本人認証としての利用されるだけではなく，近年ではショッピングモールなどで，無条件に顧客の属性を自動推定し，人流情報等を抽出する新しい応用も広がっている．その際には，本人の同意なく顔が分析され，特徴を識別されて追跡されてしまう脅威が生じる．例え，サングラスやマスクをして追跡を防止しようとしても，最近導入が著しいディープラーニング技術を用いて，マスクなどをかけたままのパターンを学習されては追跡を避けれないことが予想される．これまでのところ，この脅威に対するリスクは明らかでない．

そこで本研究では，バイザーの個人識別を抑制する技術の効果を検証するため，次の 2 つの代表的な顔認証方式を取り上げる．

1. CNN を用いて顔全体のイメージを学習する方式（CNN 方式）
2. 目，鼻，口などの顔の特徴点を抽出し，その相対座標を比較することで個人を識別する方式（特徴点方式）

これら 2 つの識別システムをそれぞれ実装し，素顔のまま学習する時と，バイザーや通常のサングラスなどを装着して学習する時とで，個人の識別精度がどのように変化するかを明らかにする．

CNN を用いた識別システムは，python の計算パッケージである NumPy[*1]と，TensorFlow のフレームワークである Keras[*2]の 2 種類の方式で実装する．顔の特徴点を用いた識別システムでは，顔の検出にコンピュータビジョン向けライブラリの OpenCV[*3]を利用し，目や口などの特徴点抽出にはクロスプラットフォームソフトウェアライブラリの Dlib[*4]を使用する．

[*1] Numpy (http://www.numpy.org/)（2019/5/11 参照）
[*2] Keras Documentation(https://keras.io/ja/)（2019/5/11 参照）
[*3] Opencv(https://opencv.org/)（2019/5/11 参照）
[*4] Dlib C++ Library (http://dlib.net/)（2018/12/20 参照）

特徴点を用いた手法 2 はバイザーによって部位を識別されることが防止され，識別精度が落ちることが予想される．一方，CNN は，バイザーをパターンとして学習するので，大きな効果に生じないことが予想される．

本研究の新規性は次の通りである．

1. 本人協力のない環境下を想定して，バイザーやサングラスごと学習して追跡されるリスクを明らかにする．
2. Dlib，CNN，バイザー，サングラスの追跡防止効果の差を明らかにする．

A.2　要素技術

A.2.1　プライバシーバイザー [1]

プライバシーバイザー [1] は，デジタルカメラの顔検出の主流手法である Viola-Jones 法 [3, 4, 5] で用いられる Haar-like 特徴量の算出を防ぐことで顔検出を防止するデバイスである．Haar-like 特徴量は顔の明暗差に着目し，２つの異なる矩形領域で構成された Haar-like を用いた特徴量である．

図 A.1 に Haar-like 特徴の基本パターンを示す．プライバシーバイザーは顔全体で最も明暗差のある目元に対して，本来の明暗と反対となるように目の周りに白い反射性素材を用い，鼻筋に黒い吸収性素材を用いることで顔検出を失敗させる．

図 A.2 に，本研究で使用するプライバシーバイザーの商用版である株式会社前澤金型の PV-001 の CL BLACK[11] を示す．本プライバシーバイザー [1] は，レンズとテンプルの接合部であるヨロイが上下に可動出来るようになっており，ヨロイを閉じた状態でサングラスの機能のみ持つ「通常モード」（以下，バイザー OFF）と，開いた状態で顔検出を防ぐ「保護モード」（以下，バイザー ON）の 2 状態を有する．

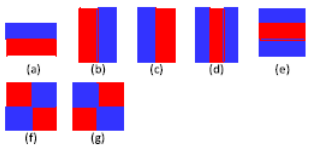

図 A.1　Haar-like 特徴の基本パターン [1]

図 A.2　本研究で使用したプライバシーバイザー [11]

A.2.2　CNN を用いた顔識別法

図 A.3 に Convolutional Newral Network(CNN) の概要図を示す．Convolutional は畳み込みを表し，画像のピクセルの座標位置に関する特徴量を学習に取り入れる．人間の脳神経系のニューロンの働きをモデル化したニューラルネットワークは，与えた学習データに対して，逆誤差伝播法により各々のニューロンが持つ重みとバイアスを適正化する．

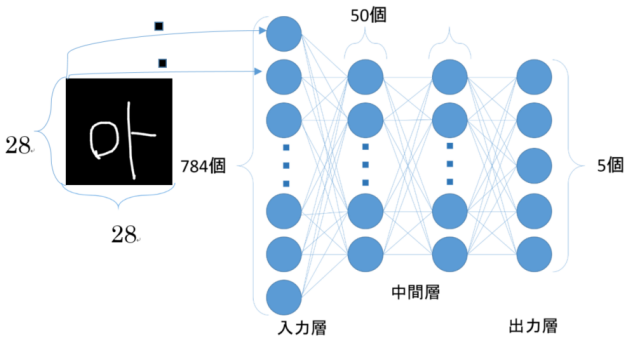

図 A.3　ニューラルネットワークの概念図 (ハングル語の母音を画素ごとに入力)

CNN の実装

本研究では Keras フレームワークを用いて，VGG-16[6] は ImageNet Large Scale Visual Recognition 2014(ILSVRC2014) の 1000 クラス識別タスクにおいて, 2 位となった識別手法を改良したモデルを用いる．ImageNet は，2009 年にプリンストン大学によって提供された機械学習のためのデータセットであり，2 万を超えるカテゴリと 1400 万枚以上の画像を有している．VGG は 13 個の畳み込み層とプーリング層，全結合層を組み合わせたシンプルな構造である．高い精度を発揮できるため様々な機械学習のモデルとして多用されている．

さらに，本研究ではニューロンをランダムに消去しながら学習する Dropout をすることで，過学習を抑制できる手法 [7] を追加した．また, 認証精度を向上するため，ImageNet の学習済みのモデルを転用して新たなモデルを生成する fine tuning を使用した．

A.2.3　顔の部位の相対距離を用いた識別方法

Dlib ライブラリ

本研究では顔特徴点の抽出に，クロスプラットフォームソフトウェアライブラリの Dlib を使用する．Dlib は 2002 年から Davis E. King ら [8] によって開発が開始され，現在では機械学習や画像処理等，幅広い分野のツールが追加されている．顔特徴点を抽出する機能は Vahid Kazemi らによって提案されたアルゴリズム [9] を用いている．

特徴点

手法 2 では，顔画像から顔領域を自動的に検出して，切り取りを行う為に，OpenCV の Cascade Classifier メソッド [10] による Haar-like 検出器を用いる．切り取った画像を 300 × 300 の画像にリサイズする．図 A.6 に正規化した顔画像の例を示す．

Dlib は顔の特徴点を 68 点取得できるが，その内，安定して映っている図 A.7 の中の青点で示す目，口，鼻，眉毛の輪郭の 16 点の画像左上を原点とする相対座標からなるベクトルを特徴量ベクトルとする．

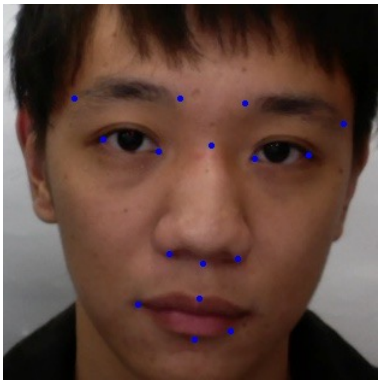

図 A.4　元画像

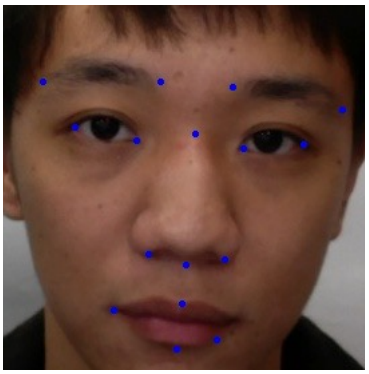

図 A.5　正規化後（300 × 300 ピクセル）

図 A.6　正規化処理の例

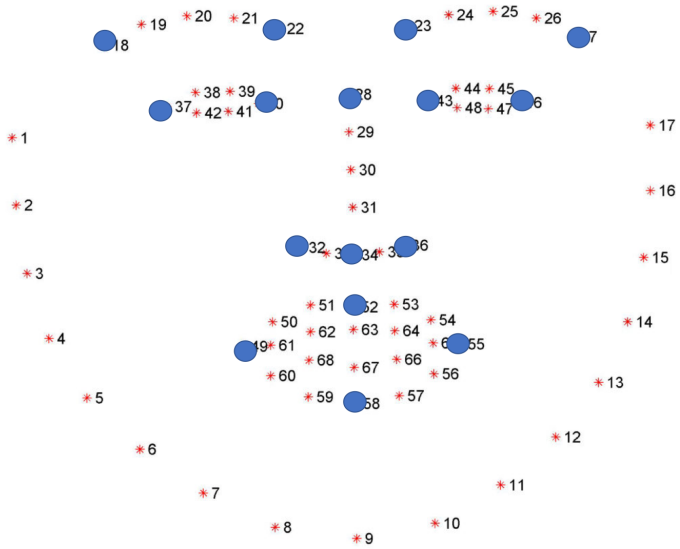

図 A.7　Dlib の抽出する部位の 68 個の座標と方法 2 で使用する特徴点

テンプレート

本実験では顔を検出することを出来た顔画像から，被験者一人あたり三枚を無作為に選択し，三枚分の特徴点の平均座標をテンプレートとしてデータベースに登録する．

テンプレートには顔検出された顔画像のみを使用するため，被験者によっては登録に必要な情報が足りない事がある．その際は，各条件で取得した全画像の特徴点の平均座標をテンプレートとして登録する．

類似度

特徴点のインデックス集合 $F = \{18, 22, 23, 27, 28, 32, 34, 36,$
$37, 40, 43, 46, 49, 52, 55, 58\}$, x_i, y_i を特徴点 $i \in F$ のテンプレートの x, y 座標, x'_i, y'_i を入力画像 A の特徴点 i の x, y 座標とする．入力画像の特徴点の座標とテンプレート B との特徴点の座標のユークリッド距離を

$$S(A, B) = \sum_{i \in F} \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2}$$

により求める．

図 A.10 は同一人物の特徴点距離と他人同士の特徴点距離を比較した例である．緑線は特徴点間の距離を表し，この総和を（非）類似度 $S(A, B)$ とする．

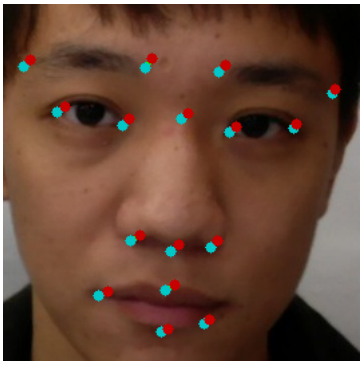

図 A.8　被験者 1 とテンプレート 1 を比べた時

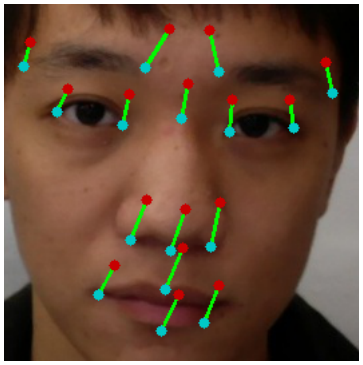

図 A.9　被験者 1 とテンプレート 2 を比べた時

図 A.10　座標の差 (赤点：テンプレート画像　青点：入力画像　緑線：特徴点の差)

識別方法

提案手法の処理の流れを図 A.11 に示す．入力画像から OpenCV ライブラリを用いて顔検出を行い，Dlib を用いて特徴点 A を抽出し，抽出した特徴点とデータベースの N 個の全テンプレートとの距離 $S(A, B_1)$, ...,$S(A, B_N)$ を計算する．A との距離が最も小さいテンプレート $\hat{k}$ を出力する．

A.2.4　適合率及び平均適合率

本研究では，顔認証の精度の評価指標として，適合率を算出する．適合率は予測したデータのうち，正しいものの割合である．例えば A さんの適合率は

$$P_A = \frac{A \text{ さんと正しく判定した数}}{A \text{ さんと判定した全ての画像数}}$$

と定義する．平均適合率は全ユーザについての適合率の平均とする．

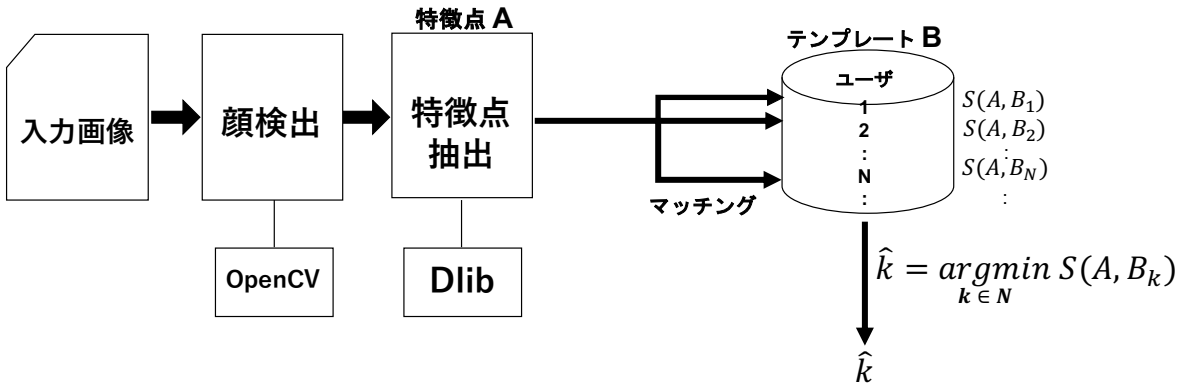

図 A.11　顔特徴点を用いた提案システムの概要

A.3　評価実験

A.3.1　実験目的

本実験の目的を以下に示す.

1. CNN と顔特徴点を用いた顔認証システムのそれぞれの精度を明らかにすること
2. プライバシーバイザーによる検出防止効果を明らかにすること
3. 素顔とバイザーごと学習した時の効果を明らかにする
4. CNN を用いた顔認証システムにおけるバイザーとサングラスの効果の比較すること

本目的の為に，素顔，バイザー OFF，バイザー ON の計 3 種類のデータで，それぞれ学習（テンプレート）をする．各学習データに対して，3 種類の評価データを使用し，評価を行う．

A.3.2　実験方法

顔画像データの取得

被験者 20 名は本学の学生 20 名（男 19 名，女 1 名）である．2018 年 7 月 7 日から 3 日間実験した．

CNN は明るさや表情, 髪形などの変化に対して汎用性を持つ必要性がある．そのため, CNN で用いる顔画像データを被験者 20 名に対し 1 日毎に 100 枚ずつ，異なる時間に顔を撮影した．顔の検出には，OpenCV を用いて取得する範囲を画面上に表示させ，図 A.12 に示すように，被験者の顔の範囲である青枠に基準となる赤枠を重ね合わせるように撮影した．iMac に標準で搭載されている web カメラを用いる．カメラの前にいる被験者を 3 フレーム毎に顔を上下左右に動かしながら撮影し, 2 日間で計 200 枚の学習データを表 A.1 に示す条件で取得した.

CNN の評価データと顔特徴点で使用するための顔画像を，顔を固定させた状態で，表 A.1 の条件で取得した.

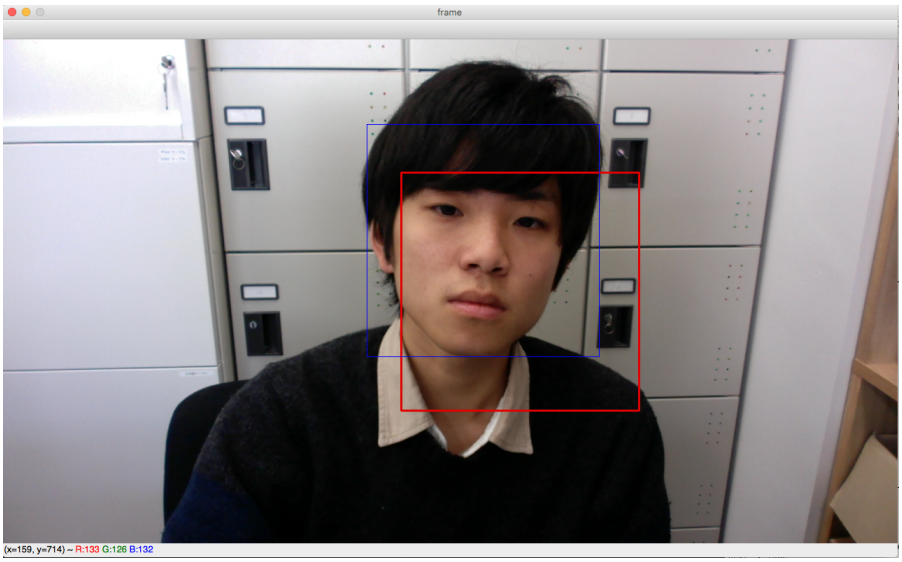

図 A.12　顔画像の取得方法

表 A.1　1 日で取得した条件別の画像数

条件	取得枚数
素顔	100 枚/日
バイザー OFF	100 枚/日
バイザー ON	100 枚/日

顔画像データの拡張

CNN を用いる場合にデータセットの数は重要な要素であり，学習データの枚数を増やすことにより CNN の精度が向上する．本研究では，Data Augmentation と呼ばれるデータ拡張を行う．元の学習データに変換を加えてデータ量を増やす手法である．取得した顔画像に対し, Keras の機能を用いて 224 × 224 にリサイズを行い, OpenCV を用いて表 A.2 に示す各パラメータについて 9 種類のデータに拡張した．

コントラストは画像の輝度の分布を考慮して，0 から 255 の大きさに調整する．

ガウシアンノイズは各画素にガウス分布に基づく乱数を足す．表 A.2 に示す標準偏差 σ を用いる．

元画像を含め, 一人一種類当たり 2,100 枚, 合計 42,000 枚の顔画像データに拡張した．図 A.19 に拡張した結果の例を示す．

表 A.2　パラメータ

手法	コントラスト調整	左右反転	ガウシアンノイズ σ
種類	12% 23% 35% 47% 59% 70%	1	2 4
計	6	1	2

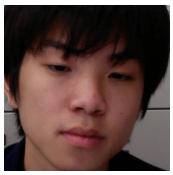

図 A.13　元画像

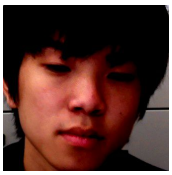

図 A.14　コントラスト 47%

図 A.15　コントラスト 70%

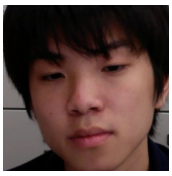

図 A.16　左右反転

図 A.17　ガウシアンノイズ $\sigma = 2$

図 A.18　ガウシアンノイズ $\sigma = 4$

図 A.19　画像の変換例

実験 1:CNN を用いた方式の実験方法

学習データを 44000 枚，テストデータを 2000 枚に分ける．学習パラメータは表 A.3 に示す．

表 A.3　CNN の構成表 (活性化関数及び損失関数は vgg16[6] に基づく)

パラメータ名	値
モデル	vgg16+Dropout
入力画素	224 × 224
バッチサイズ	128
エポック数	10
出力層	20

実験 2：顔特徴点を用いた方式の実験方法

被験者 20 名に対し，それぞれ条件で 100 フレーム分 20 × 3 × 100=6000 の画像を取得する．

実験 3：サングラスとバイザーの比較実験

本研究で使用するプライバシーバイザーと通常のサングラスに追跡防止効果の違いがあるのか比較する．サングラスを用いた実験結果 [2] と本研究の結果を比べる．

A.3.3　実験結果

実験 1：CNN による平均適合率

3 種類の顔画像を学習させた CNN の平均適合率を表 A.4 に示す．素顔を学習させた場合，素顔の評価データについての適合率は 62.59% と最も高い．しかし，バイザー ON で評価した時，平均値は 62% が 20.75% にまで下がっている．

バイザー OFF，ON を学習させた時は，評価画像のバイザー状態がどちらであっても平均適合率は 40% 以上であった．

表 A.4　方法 1：CNN を用いた方式の平均適合率

評価 / 学習	素顔		バイザー OFF		バイザー ON		平均	
	μ	σ	μ	σ	μ	σ	μ	σ
素顔	**62.59**	0.40	36.67	0.43	20.75	0.38	40.00	0.40
バイザー OFF	27.36	0.36	**59.37**	0.37	44.78	0.39	43.80	0.37
バイザー ON	40.34	0.41	**42.58**	0.34	42.28	0.38	41.73	0.38
平均	43.40	0.39	46.21	0.38	35.90	0.38	41.84	0.38

実験 2：顔検出率と顔特徴点による平均適合率

OpenCV により正しく顔を検出できた割合を表 A.5 に示す．顔検出率は正しく検出できた評価データの割合である．

3 種類の顔画像をそれぞれテンプレートとして評価した平均適合率を表 A.6 に示す．素顔の画像をテンプレートとして，素顔のデータで評価した時の，適合率は 71.17% と最も高い．バイザー OFF，バイザー ON それぞれの適合率は，6.21%，8.74% である．バイザーが素顔のテンプレートに対して効果を示した．

テンプレートをバイザー OFF，ON にし，同じ条件下で評価を行った際，評価データも同じくバイザー OFF，ON とした適合率は 48.01%，49.16% と最も高い結果になった．しかし，バイザー OFF をテンプレート，評価をバイザー ON で行った際の適合率は 15.29%，テンプレートをバイザー ON，評価をバイザー OFF で行った際の適合率は 16.37% と，バイザーの状態を変えることで適合率は大きく下がる．

表 A.5　種類別の顔検出率

	顔検出率 [枚/2000 枚]
素顔	1774 (88.7%)
バイザー OFF	1391(69.5%)
バイザー ON	1198(59.9%)

素顔を学習させた時のユーザ毎の認証精度の変動

二つの提案システムで外乱に対する個人差に違いが出るのか見るために，素顔を学習させた時の被験者毎の素顔（X 軸）とバイザー ON 時（Y 軸）の平均適合率の散布図を図 A.20 に示す．

CNN において，被験者によっては素顔を学習した際の汎用性に違いを示した．被験者 1，2，3 は，バイ

表 A.6　方法 2：顔特徴点を用いた方式の平均適合率

評価 / 学習	素顔		バイザー OFF		バイザー ON		平均	
	μ	σ	μ	σ	μ	σ	μ	σ
素顔	**71.17**	0.25	6.21	0.09	8.74	0.20	28.71	0.18
バイザー OFF	16.41	0.30	**48.01**	0.26	15.29	0.24	26.57	0.27
バイザー ON	8.95	0.25	16.37	0.25	**49.16**	0.31	24.83	0.27
平均	32.18	0.27	23.53	0.20	24.40	0.25	26.70	0.24

ザーを付けても適合率の低下が生じていない．すなわち，バイザーによる追跡の抑制効果が小さい．しかし，ほとんどの被験者はバイザー装着によって適合率が 0 に下がり，素顔の評価も 10% 以下か 90% 以上と二極化する結果となった．その理由として，被験者一人一人の画像に多様性がなかったため，CNN が素顔を学習した時の適合率が 0% に近い被験者と 100% に近い被験者の二極化されてしまった．したがって，静止の顔画像を取得する際は 1 枚毎に一定の期間を空ける分散撮影をすることが求められるだろう．

次に，顔特徴点による識別手法（Landmarks）の認証精度の変動について述べる．CNN と同様，バイザー装着によってほとんどの被験者の適合率は 0 に下がったが，被験者 4 のみ例外的に減少が見られなかった．素顔に対する評価は 20% ～ 100% でばらつきがある．

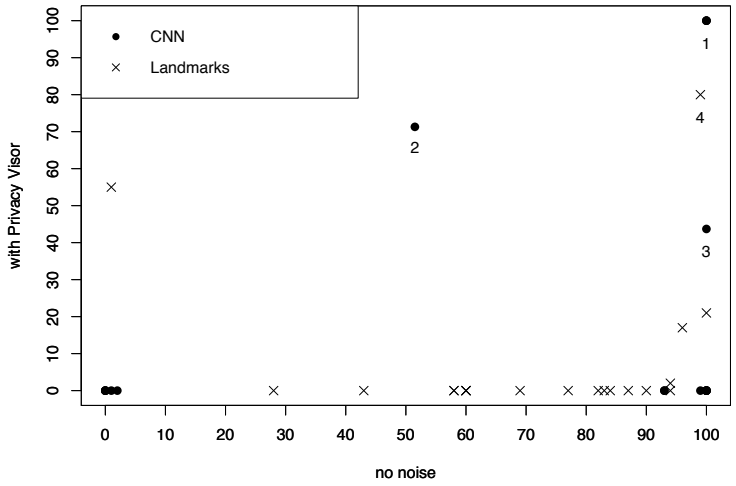

図 A.20　素顔を学習した際の各ユーザの認証精度の変動（X 軸：素顔の評価データの適合率，Y 軸：バイザーの評価データの適合率）

バイザー ON を学習させた時のユーザ毎の認証精度の変動

前節同様に，バイザー ON を学習させた時の被験者毎の素顔（X 軸）とバイザー ON 時（Y 軸）の平均適合率の散布図を図 A.21 に示す．

バイザー ON の顔画像を学習した CNN において，被験者の適合率は全体的に上に位置している．すなわち，バイザーの追跡抑制効果が低いことを示している．興味深いことに素顔とバイザー ON のどちらについても適合率が高い被験者は図 A.20 よりも多いことから，素顔よりバイザー ON で学習した方が汎用性がある．

顔特徴点を用いた手法において，素顔をテンプレートとした時と同様に 5，6 番以外のユーザは素顔の評価が 15% 以下であり，CNN と比較して汎用性が低いことを示した．

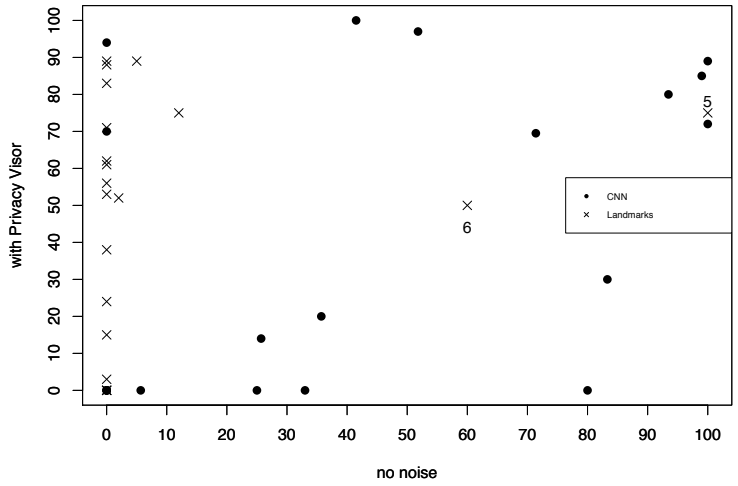

図 A.21　バイザー ON を学習した際の各ユーザの認証精度の変動（X 軸：素顔の評価データの適合率，Y 軸：バイザーの評価データの適合率）

実験 3：比較実験の結果

2018 年 3 月 15 日にサングラス，マスク，帽子などの外乱に対して，顔認証の精度が低下する効果を報告した [2]．表 A.7 に本稿との違いを示す．

表 A.4 より，学習と評価を同じバイザーで評価した場合，平均適合率が 42.28% となったのに対し，一般的なサングラスを用いて評価した [2] では，94.4% と非常に高い精度であった．

表 A.7　本研究と [2] の比較

評価時期	2017	2018
手法	独自 CNN(vgg-11)	keras(vgg-16)
外乱	帽子，マスク，サングラス	プライバシーバイザー
平均適合率	**54.80**	**41.73**
外乱頑強性	低い	高い

A.4　考察

A.4.1　CNN によるプライバシーバイザーの評価

表 A.4 より，素顔を学習し，素顔で評価した場合の適合率が最も高くなったのは自然である．一方，バイザーで評価した場合は，後述するレンズの特性により，適合率が低下したと考える．バイザーはミラー型のレンズを使用しているため，バイザーに反射する景色が CNN に影響を与える．反射する景色は，被験者の顔の傾き等の個人の特徴が影響する場合と，研究室内の人の移動等の外部の環境変化が影響する二通りがある．

一方，一般的なサングラスはグレーに着色された色付きレンズを使用しており，レンズ部の光反射が無いため CNN への影響が小さい．サングラス以外の露見している部分に対して細かく学習したことで精度を高めたと考える．

A.4.2　顔特徴点によるプライバシーバイザーの評価

バイザーを掛けた顔画像は顔特徴点を正しく取得できないため，バイザーでの評価が低下したと考える．

しかしながら，追跡を完全に防止できたわけでない．例えば，表 A.6 より，バイザー OFF，バイザー ON をテンプレートとし，同じ外乱の評価データの適合率は 48.01%，49.16% であった．すなわち，バイザーをかけても追跡をされるリスクが残っていることを示している．

A.4.3　ユーザ毎の変動評価

図 A.20 により，被験者によって素顔を学習した時のバイザーへの汎用性の違いがみられた原因として，目元の特徴量への依存が考えられる．すなわち，CNN は被験者それぞれに対して，その被験者を識別する顔の特徴的な部位を学習し，判断しているので，目以外の特徴で識別できたと考える．

一方，図 A.20 と図 A.21 により，顔特徴点を用いた識別システムは比較的汎用性が低い．これは取得した特徴点が一つでもずれてしまうと類似度に大きく影響するからだ．

A.5　おわりに

バイザーを装着することにより，CNN を用いた顔認証による個人識別の平均適合率を素顔の場合と比べ 26%〜42% 下げることを明らかにした．その減少の効果は認証アルゴリズムに依存し，例えば顔特徴点を用いた顔認証においては 63%〜66% まで下げる．しかし，バイザーを装着したままで学習すると，本人と識別される割合が高くなる．特に CNN においてはバイザーを学習した場合，評価が異なる条件下でも 27.36% 以上の適合率であった．つまり，CNN を用いた顔認証システムは頑強性がある．そのため追跡防止を目的にバイザーを装着したとしても，バイザー装着時の画像をデータベースに登録されてしまうと追跡をされてしまうリスクがある．

本研究では，提案システムの学習や評価に正面からの顔画像を使用した．監視カメラから取得される，遠くから様々な角度で写る画像での評価を今後の課題とする．

参考文献

[1] 山田隆行，合志清一，越前功“光の反射・吸収特性を利用した撮影画像からの顔検出防止手法”，情報処理学会論文誌 Vol.55，No 9，pp.2104-2119，2014.

[2] 脇一史, 菊池浩明，“CNN を用いた顔認証システムの開発と追跡停止に対する評価”，情報処理学会第 80 回全国大会，7W-03，pp.3-543-3-544，2018.

[3] Viola，P. and Jones，M.: Rapid object detection using a boosted cascade of simple features，Proc. Computer vision and Pattern Recognition 2001 (CVPR 2001)，pp.I-511-I-518，2001.

[4] Shakhnarovich，G.，Viola，P. and Moghaddam，B.: A unified learning framework for real time face detection and classification ，Proc. Automatic Face and Gesture Recognition 2002 (FG2002)，2002.

[5] Viola，P. and Jones，M.: Robust Real-Time Face Detection，International Journal of Computer Vision(IJCV)，Vol.57，No.2，pp.134-157，2004.

[6] Karen Simonyan and AndrewZissserman(2014)，“Very Deep Convolutional Networks for Large-Scale Image Recognition”，pp.1409-1556，ICLR，2014.

[7] 齋藤康毅，“ゼロから作る Deep Learning python で学ぶディープラーニングの理論と実装”，OREILLY，2016.

[8] King, D. E. (2009)，“Dlib-ml: A Machine Learning Toolkit”. J. Mach. Learn. Res. Vol.10 (Jul): 1755-1758，2009.

[9] Kazemi, V., Sullivan, J.，“One Millisecond Face Alignment with an Ensemble of Regression Trees”, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)，pp. 1867-1874，2014.

[10] Python OpenCV の基礎 ついに顔検出してみます - Python の学習の過程とか (http://peaceandhilightandpython.hatenablog.com/entry/2016/02/18/194303，参照 2019/5/11).

[11] PRIVACY VISOR (http://www.privacyvisor.jp/，参照 2019/5/11).