

画像認識技術を用いた三種の開発型研究

高橋 拓 井田 翔琉 池上 劉虎

群馬県立高崎高等学校

〒370-0861 群馬県高崎市八千代町2丁目4番1号

1. 研究の背景

現在、画像技術は多くの場面で用いられ、研究が進んでいる。そこで、画像技術を用いた開発を行い、独自の利点を持つシステムを作ろうと考えた。開発したシステムはそれぞれ ハンドサインをもとにした認証(以下surpass-words)、顔認証と移動方向推定を組み合わせた在宅管理システム(以下housensor)、OCRと正規表現による料理の分量計算を行うシステム(以下instant calculator)である。

2. 研究の目的

2.1 研究全体の目的

インターネット上には数多くのオープンソースコードが存在し、誰でも高度な技術を活用することができる。そこで、その組み合わせとコードの工夫により既存の製品と差別化を図り、新たな製品ないし試作品を開発しようと考えた。surpass-wordsは新たな認証の開発、housensorは高齢者の徘徊の防止、instant calculatorは素早い分量計算を目的として研究を開始した。

2.2 仮説

既存の製品を分析し、差別化を図れる点を見出せば、優位性を保って開発が行えるのではないかと考えた。また、Googleが様々なソースコードを提供しているため、Googleのコードに変更を加えてプログラムを作成できると考えた。

3. 実験方法

3.1 調査

開発したい製品の分野を調査し、既存のサービスや製品の特徴を調べた。現在の認証システムとして、顔認証、虹彩認証、静脈認証などの生体情報による認証、鍵、ICカードなどの所持情報による認証、パスワード、PINコードなどの知識情報による認証の主に3種類が存在している。特徴として、生体認証、所持認証は安全性は高いものの、変更や共有が難しく、知識認証は変更共有は容易であるが、安全性の面では他の認証システムに劣ることが挙げられる。在宅管理システムには様々な種類が存在している。特徴としては、ブザー一つで異常を知らせることができる、駆けつけ対応があり有事の際に迅速な対応が可能であることが主に挙げられる。顔による管理はされていないことが判明した。料理計算としてはまとめて計算が可能なアプリケーションが存在している。特徴は、それぞれ設定した分量に対して全体に計算が行えること、インターネット上にレシピを登録できることが挙げられる。数字入力の手作業であることが確認できた。

3.2 方法

全てpythonを用いてプログラムを作成した。

surpass-wordsはwindowsのコマンドプロンプト上で実行した。コードはgoogleの提供するMediaPipe... (1)上のものを参考にした。まずMediaPipe handsで手のランドマークを取得し各指が曲がっているか伸びているか判別し、手の状況を32通りに分類した。これにより手の形を一桁32通りのパスワードと

して使用可能になった(I)。事前に設定した何桁かのパスワードと照合し、認証に成功すると群馬大学のサーバーに信号を送るよう設定した。信号が送られた後、群馬大学のサーバーからソレノイドロック(5V 0.8A 通電1000ms)に接続したesp32に信号を送り、解錠されるようにした。同時に 圧電ブザーで解錠したことを通知するようにした。

他にも、housensorはRaspberry Pi上に作成し、カメラを取り付けた。まず、face recognition...(2)による顔のランドマーク取得(II)及び検出した顔のランドマークの比較を行った。比較の結果Toleranceが3未満のものを適合とした。その後、検出した人物をpeople counterを利用して追跡した...(3)。撮影した動画上に境界線を設定し、通過した人の移動方向を二次元座標の変化から判定した。適合した写真の元データの名称と、移動方向から玄関ドアの出入りをSlackに通知するよう設定した。

instant calculatorはGoogle Colaboratory上に作成した。まず、料理のレシピの分量の部分をカメラで撮影するか機器内に保存された写真を取得することにより元データとなる写真を設定した。続いて、Tesseract 4(以下OCR)により画像から文字を取得した。その後、正規表現により、数字と小数点のみを抽出し、連続した部分を一つのデータとして取り出した。データをリスト化し、掛け算を行うことで数値を調整できるようにした。

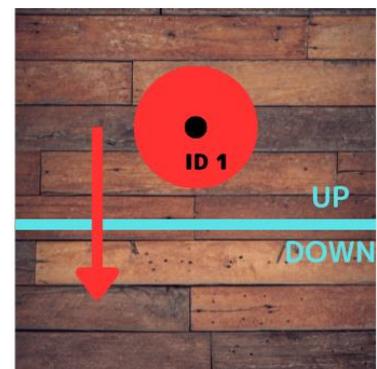
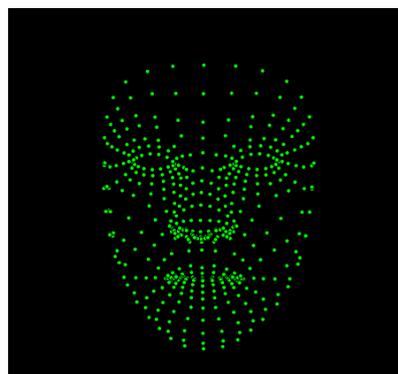
4. 結果と考察

4.1 結果

surpass-wordsでは手の形の判別を高精度で行うことができた。パスワードの認証に成功してから実際に解錠されるまで少しタイムラグが生じてしまった。また、ハードウェアの問題で解錠が正常に行われない時があった。

housensorでは顔を判別した後に続けてドアの出入りの判定する一連の判定を正確に行うことができた。一人の顔を判別する際に平均して10秒程度の時間がかかった。出入りの判定は1秒で実行が完了した(III)。二人以上がカメラに写った際には判定を行うことができなかった。

instant calculatorでは数値計算を正確に行うことができた。OCRはtesseract 4の精度と同じであり、活字での実行で数字抽出におけるエラーは見られなかった(IV)。



(I)ハンドサイン認証の様子

(II)顔のランドマーク取得

(III)移動判定イメージ

入力の順番と手の形が表示される
仕組みとなっている。数字は
親指から小指にかけて順に
2進数で置き換えている。

顔を点に置き換えている。
点同士の距離を三次元的に
算出し、動画に映る顔の点
と比較している。

判定の線を上下のどちらから通ったかを判定
している。各人にはID
が自動でつく。

📄 パスタ 500g
オリーブオイル or バター am
玉ねぎ 1.25
ピーマン 5 個

(IV)OCRの結果

漢字に関しては多少のミスがあったものの、
数字に関してはミスが見受けられなかった。

4.2 考察

surpass-wordsでは一度群馬大学のサーバーを経由しているため、タイムラグが起こってしまったと考えられる。また、現段階ではwindowsでの実行しかできていないため、利便性のため今後は小型化、軽量化を進めて、どのような場所でも使用可能にする必要がある。他にも、パスワードを他者に見られない工夫をする必要があると考えられる。

housensorでは顔判定時の計算処理が多いことが判定の時間を伸ばしていると考えられる。先行研究のnekobean...(4)によるとCPUによる演算はGPUに比べ処理が遅くなることがわかっており、CPUのみの対応であるChromebookを利用していたことが一因である可能性がある。また、二人以上検出できないことについても、プログラムの性質上一人を対象として比較を行うために、エラーが発生したのだと考えられる。ただし、玄関に同時に二人以上いる場合、徘徊を防止するという目的を果たす上で支障は無いと考えられる。

housensorの研究を行う上で、CPU対応のChromebook上で開発をしており、GPUを用いた資料が多かったために参考資料を発見することが困難であった。画面上に描画をすることも不可能であり、文字による描画で対応せざるを得なかったことも研究を困難にした。自前のパソコンを用意することでこの問題は解決できると思われる。

instant calculatorでは十分な精度でOCR及び計算が行われたと思われる。一方で、webアプリ化がまだなされておらず、広く使われるにはアプリ化が必須であると考えられる。OCRによる分量計算アプリが存在していない理由として、写真の取り方による精度のばらつきが考えられる。写真のブレや部屋の明るさにより精度は変動するため、手法として手打ちによる計算が用いられていることも考えられる。一方で、早さや手軽さの観点からすると、このinstant calculatorは既存のサービスに勝るものであると考えられる。

5. 結論

5.1 結論

surpass-wordsは新たな認証の開発という点では目的を達成したが、実用化にはまだ多くの課題がある。娯楽としての用途や、水を用いる場面での認証としては一定の効果があると考えられる。

housensorは、処理の高速化により十分活用できるものになると考えられる。これには開発環境の変更やコードの書き換えが想定される。

instant calculatorは一連の処理をwebアプリ上で行えるようにすることで利便性が向上すると考えられる。

開発全体としては、インターネット上のオープンソースコードを活用し、実際に独自の利点を持つサービスを開発することができた。また、開発をする中で既存のサービスの利点を深く理解するとともにpythonやlinuxに対する理解を深めることができた。

5.2 展望

本研究で開発した諸サービスの利便性の向上のため、webアプリ化をすることが今後の目標である。また、フィードバックを重ねることで利用者に寄り添ったよりよいサービスになると考えられる。

6. 参考文献

- (1) google. “Hand landmarks detection guide”.MediaPipe.2023-06-09.
<https://google.github.io/mediapipe/solutions/hands.html>
- (2) Yuya Kato. “InsightFaceで顔認証(特微量抽出、比較)してみた”.Zenn.2021-8-14.
<https://zenn.dev/yuyakato/articles/d35b185d36a33b>
- (3) Adrian Rosebrock. “OpenCV People Counter”.pyimagesearch.2018-8-13. https://pyimagesearch-com.translate.goog/2018/08/13/opencv-people-counter/?_x_tr_sl=en&_x_tr_tl=ja&_x_tr_hl=ja&_x_tr_pto=sc
- (4) nekobean. “Python - 顔認識ライブラリ Face Recognition で顔認証を行う方法”.Pystyle.2020-02-26.
<https://pystyle.info/perform-face-recognition-with-python/>

謝辞

群馬大学の中村先生や株式会社ペリテック井澤様をはじめとするメンターの皆様, 鈴木先生, 岡田先生, この研究を行うにあたってご助力を頂いたすべての方々に感謝申し上げます。