



# アバター表現のための瞬目種類識別

Blink type identification for avatar representation

早川要<sup>1)</sup>, 関庸一<sup>2)</sup>, 青山一真<sup>2)</sup>, 松野省吾<sup>3)</sup>

Kaname HAYAKAWA, Shogo MATSUNO, Kazuma AOYAMA, and Yoichi SEKI

1) 群馬大学大学院 理工学府 理工学専攻 電子情報・数理教育プログラム (〒 371-0044 群馬県前橋市荒牧町 4-2, t231d060@gunma-u.ac.jp)

2) 群馬大学 情報学部 (〒 371-0044 群馬県前橋市荒牧町 4-2)

3) 電気通信大学 大学院情報理工学研究科 (〒 182-8585 東京都調布市調布ヶ丘 1-5-1)

**概要:** 本研究では, HMD に搭載されたアイトラッカーによって検出された瞬目が随意性か否か自動で識別する手法を検討する. 筆者らは仮想空間での高粒度なコミュニケーションを行うために, ユーザの心理状態を非随意的な生理信号から推定し, アバターの動作や感情表現を心理状態と連動させるシステム構築を目指している. その基礎的な検討として覚醒状態の推定に用いられる指標である瞬目の生起から, アバター表現にフィードバックするシステムを検討し, その結果を報告する.

**キーワード:** アイトラッキング, 瞬目識別, アバター表現

## 1. はじめに

仮想空間においてはソーシャル VR など, アバターを介して相手とコミュニケーションをすることができる. しかしヘッドマウントディスプレイ (以下, HMD) を通して精密に心理状態を測定し, 現実でのやり取りのように表情の機微など表現しながらやり取りを行うことは困難である. より容易に測定できるデータから心理状態を推定し, アバター表現へ利用できるシステムがあれば, 広く利用されるシステムとなることが期待できる.

近年ではアイトラッカー内蔵の HMD が商用販売されており, VR のアプリケーションを実行しながら眼球の状態を測定し入力チャンネルとしてアプリケーションを操作することも可能になっている. 筆者らは, HMD 内蔵のアイトラッカーから測定できるデータを用いて心理状態を推定し, アバター動作や感情表現へ利用することを考えている. 本稿ではその基礎的な検討として, 覚醒状態の推定に用いられる瞬目を検出し, その種類を識別してアバター表現にフィードバックするシステムを検討する [1]. また, HMD から API(Application Programming Interface) を通して取得できるデータは測定時の目の画像などの生データを取得できるとは限らず, どのような VR の実行環境でも応用できるようにするため, 入手しやすいと考えられる測定時の「瞼の開き具合」のみのデータに注目したシステムの検討を行う.

## 2. 非随意性瞬目の検出

### 2.1 検出手法の概要

随意性瞬目と非随意性瞬目の含まれたデータから瞬目を種別し, 覚醒状態の推定に用いられる非随意性瞬目を識別

する. 瞬目の検出には, 片方の瞼の開き具合を示すデータを用いる. このデータは開眼時に 1.0, 閉眼時に 0.0 となるように正規化されているものとする. この瞼の開き具合の時間的な波形をプロットしたものを瞬目波形と呼ぶ. 瞬目波形は図 1 のような波形となる. 瞼の開き具合のデータが測定されたフレーム毎に, 前フレームとの差分を取り, フレーム間の時間で割ることで離散時間微分係数が計算できる. 微分係数が負になったフレームを瞼の閉じ始めとし, 負または 0 から正になったフレームをピーク, 正から 0 または負になったフレームを瞼の開け終わりとし, この一連の連続したフレームを一回の瞬目として抽出する.

ここで検出される瞬目は, 目の開閉がすべて検出されることとなるため, 随意性瞬目, 非随意性瞬目, そして瞼の痙攣や計測誤差を含むノイズが検出される. 随意性瞬目と非随意性瞬目のうちの一つである自発性瞬目の間には, 瞬目時の振幅や持続時間に差が見られることが示されている [2][3]. 図 2 に示すような連続した瞬目が得られたとき, 振幅  $A$  は, 閉瞼時振幅  $A_{close}$  と開瞼時振幅  $A_{open}$  のうちより大きいほうの値とし, 持続時間  $time$  は瞬目全体のかかった時間を示す. これらのパラメータについて, キャリブレーションを行い閾値を決めることで, 随意性瞬目, 非随意性瞬目, ノイズの識別を行う.

### 2.2 閾値の決定

リアルタイムに検出された瞬目を種別するための閾値を決定するため, システムを使用するユーザのキャリブレーションを行う. キャリブレーションから得られた瞬目の振幅, 持続時間のデータから, 随意性瞬目, 非随意性瞬目を識別する前に, 最初にノイズを除去する. まずはデータのノ

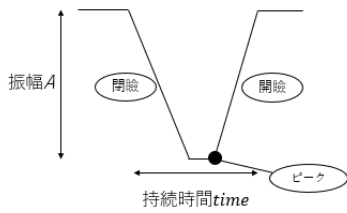


図 1: 瞬目波形のイメージ図

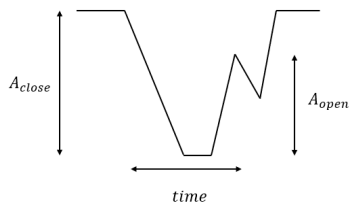


図 2: 連続した瞬目のときの瞬目波形

イズを軽減するため、範囲を 5 に設定した移動平均フィルターを適用し平滑化処理を行った。その後、振幅パラメータ  $A$  に対し K-means 法でクラスタ数 3 を指定しクラスタリングを行う。その結果で最も平均の小さいクラスタの中で最も大きい振幅の値を  $A_{noise}$  とし、これを下回る振幅のデータをノイズと識別して除去する。

除去した後のデータには随意性瞬目と非随意性瞬目が含まれるため、これらのデータを識別するために振幅  $A$  と持続時間  $time$  の 2 つのパラメータを使い、ロジスティック回帰を行う。2 つのパラメータでのロジスティック回帰から求められる決定境界は以下のような式になる。

$$\beta_0 + \beta_1 * A + \beta_2 * time = 0 \quad (1)$$

左辺の値が 0 以上の場合随意性瞬目、0 未満の場合非随意性瞬目と識別する。キャリブレーションデータから係数  $\beta_0, \beta_1, \beta_2$  を求め、瞬目が検出されたら振幅と持続時間の値をこの式に当てはめることで、随意性瞬目と非随意性瞬目を識別する。

### 3. システムの実装

#### 3.1 システムの概要

ヘッドマウントディスプレイには、HTC 社の VIVE Pro Eye を用いた。SteamVR を通して PC と接続し、視線追跡データは SRanipal SDK を通して取得した。アバターを表示するアプリケーションの作成および再生、データの取得には Unity を用いる。眼の開き具合のデータには、SDK の `SRanipal.Eye.GetEyeOpenness` メソッドを用いて左目のデータを取得する。このデータは開眼時に 1.0、閉眼時に 0.0 に正規化されており、80~90fps で毎フレーム取得され、アプリケーションが再生されてからの経過時間がフレームごとに記録される。

実際にこのシステムが適切に動作するかどうかを評価するための筆者自身での検証を行った。キャリブレーション

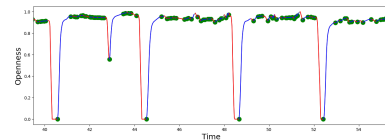


図 3: 振幅の時系列変化例

データとして、VR アプリケーションを再生し、8 秒に 1 度聴覚的な合図があったタイミングで随意的に瞬きをしたものを用いた。随意的な瞬目の際には同時に特定のキーを押すアノテーション作業をすることで、随意性瞬目の正解データを取得した。検出された瞬目数は合計で 1041 個、瞬目時にアノテーションした随意性瞬目は 39 個であった。

今回取得されたキャリブレーションデータの一部の瞬目波形を図 3 に示す。この図では、検出された瞬目のうち、閉眼時を赤線、開眼時を青線、瞬目のピークを緑点で示している。

#### 3.2 ノイズの除去

まずはノイズを取り除くため、瞬目ごとの振幅パラメータ  $A$  に対して K-means 法でクラスタリングを行った結果を図 4 に示した。クラスタ平均の最も小さいクラスタをノイズとして除去する。また、赤い点線はノイズクラスタのうち最も振幅の大きかった点を閾値とする。今回のデータでは約 0.23 であった。全瞬目数 1041 個中、ノイズに該当するデータは 980 個であった。

#### 3.3 随意性瞬目と非随意性瞬目の識別

次に、ノイズを除去したデータについて随意性瞬目と非随意性瞬目の識別を行うため、振幅  $A$ 、持続時間  $time$  の 2 つのパラメータを説明変数として使い、ロジスティック回帰を行って決定境界を計算した。アノテーションされた随意性瞬目 39 個、それ以外の 22 個の非随意性瞬目と考えられる瞬目を正解データとして扱い、うち全データの 75% を決定境界の算出に、25% はデータに対するこのモデルの当てはめの評価に使った。図 5 に決定境界をプロットと評価データの分類を示した。今回のデータでは式 (1) の係数は、 $\beta_0 = -2.52$ ,  $\beta_1 = 1.33$ ,  $\beta_2 = 2.48$  となった。

またモデルの評価結果について、随意性瞬目については適合率 0.85、再現率 1.00、F 値 0.92、非随意性瞬目については適合率 1.00、再現率 0.60、F 値 0.75 であった。正解率は 0.86 であった。全体的に高い評価スコアが得られ、随意性瞬目と非随意性瞬目の決定境界は概ね正しく算出できていると評価できた。

#### 3.4 アバター表現への応用

この研究では、VR アプリケーション上でのやり取りに瞬目などの生体情報を利用しフィードバックすることを考えている。どのようなフィードバックが効果的であるかは今後検討する必要があるが、VR 開発環境として最も利用されている Unity で結果をフィードバックできる簡単なシステムを試作しておくことで、今後の研究開発を促進できると考えられる。よって本稿では瞬目が検出、種別されたときにユーザが視認できる形でその結果の簡単なフィード

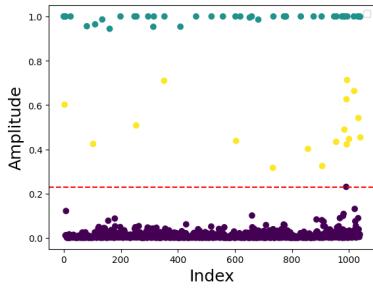


図 4: 振幅のクラスタリング結果

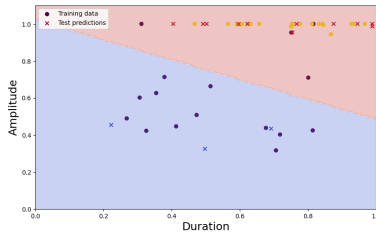


図 5: ロジスティック回帰結果

バックを行う。

Unity のシーン上にアバターを配置し、実行中に瞬目が検出されたとき、非随意性瞬目と識別された場合にはアバターの左手、随意性瞬目と識別された場合にはアバターの右手を 1 秒間赤くハイライトするように設定する。瞬目の識別には予め取得できている閾値を用い、まずはノイズを識別し、次に非随意性瞬目と随意性瞬目の識別を行う。その識別結果を用いて、アバターへのアクションへ応用する簡易的なシステムを作成した。

実装した結果、システムは正常に作動し、無意識下で行われた瞬目が検出された際には図 6 のように左手、随意的に瞬目が検出された際には図 7 のように右手が赤くハイライトされることが確認できた。これにより、VR アプリケーション実行中の瞬目を自動で検出、種別し、その結果をユーザにフィードバックすることが可能になった。今後はこのシステムを定量的に評価する実験を行うとともに、このシステムを基盤に非随意性瞬目の生起頻度からアバター表現へのフィードバックを考えるなど、より効果的に心理状態を推定しアバター表現に利用する方法等を検討する。

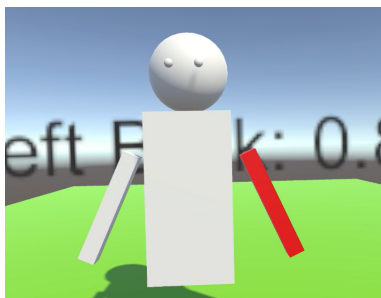


図 6: 非随意性瞬目検出時のアバターの反応

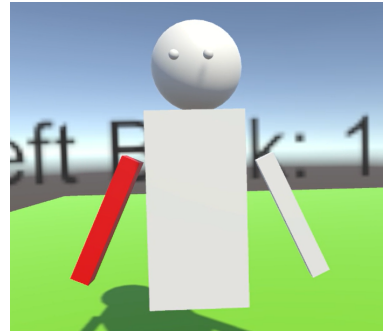


図 7: 随意性瞬目検出時のアバターの反応

#### 4. 考察

VR アプリケーションを実行中に瞼の開き具合のデータを計測しリアルタイムに検出した瞬目について、キャリブレーションをして閾値を予め決定しておき、瞬目の種別を行いアバター表現にフィードバックするシステムを作成した。ノイズ除去後の瞬目の種別については、高い正解率を達成し、識別アルゴリズムの有効性が証明できた。また、アバター表現への応用については、リアルタイムに種別された瞬目を用いてアバターにフィードバックするシステムを実装した。

しかし、ノイズの除去において今回は測定時の目の状態を確認できず、閾値決定法として用いた K-means 法の妥当性の評価ができていない。また、非随意性瞬目の識別において、アノテーションできていない随意性瞬目が非随意性瞬目の正解データとして扱ってしまっている場合があり、キャリブレーションの設計に改良の余地がある。

今後は本稿での検討を定量的に評価できるよう実験するとともに、アイトラッカーから得られる瞬目以外の運動の検出に注目し、心理状態の推定についての研究を進める。また、推定された心理状態のアバター表現へのフィードバック方法についても、より効果的なものを検討する予定である。

#### 参考文献

- [1] Bar-Kalifa, I., Plotnik, M., DeLibe, I., Ben-Zur, H., Bar-Kalifa, I.: Spontaneous eye blink rate: An index of dopaminergic component of sustained attention and fatigue, *International Journal of Psychophysiology*, Vol. 123, pp. 58–63, 2018.
- [2] 松野省吾, 阿部清彦, 佐藤寛修, 大井尚一, “随意性瞬目と自発性瞬目の識別に関する検討,” 第 11 回情報科学技術フォーラム講演論文集, 第 3 分冊, pp.23-26, 2012.
- [3] Karni, A., Benjamin, J., Tanne, D., Sagi, D.: The acquisition of perceptual and motor skills: A memory system in the adult human cortex, *Perceptual and Motor Skills*, Vol. 22, Issue 1, 2005.