

まばたきによるコミュニケーション

古川 勇 樹*・苫米地 宣 裕**

Communication by Blinks

Yuuki KOGAWA* and Nobuhiro TOMABECHI**

Abstract

In this paper, a novel communication method using blinks is presented which attempts to support the handicapped persons unable to speak and move the body by oneself. The study is composed of following two parts. (1) the face to face communication by blinks, (2) the text generation by blinks. At first, the blink characters are defined. The opening and/or closing of both eyelids and their duration time are encoded into 8 basic patterns. Then, the time sequences of basic patterns are assigned to the characters. The assignment refers to the Morse signals. It is confirmed that the face to face communication by blinks can be successfully realized without any voice or gestures. Next, the recognition of blinks using a camera and the image processing technology is studied. It is also confirmed that the receiving of blink signals and the conversion to a text can be successfully carried out.

Keywords: Blink, Handicapped-person, Communication, Blink signals

1. ま え が き

現在、身体障害者や高齢者が特別視されることなく、社会活動に参加し、健常者と共に社会へ貢献できる環境整備の促進が求められている。

口の聞けない障害者と意思疎通を図る方法として、よく使われるものに手話がある。他にも目の見えない障害者とは点字を使ってコミュニケーションを図ることが可能である。ただ、これらの方法は体全体、あるいは腕などを動かし相手と意思疎通を図る方法のため、筋萎縮性側索硬化症 (Amyotrophic Lateral Sclerosis, 通称 ALS) のような重篤な筋萎縮性障害者には用いることができない。

これまでには、首振りを用いた方法 [1] や、まばたきや歯軋りによってコンピュータ画面上

の光点の位置を指示する方法 [2] などが報告されているが、これらは専用のシステムが必要という問題点もある。

そこで本研究では、左右のまぶたの開閉 (まばたき) を符号化して、相手とコミュニケーションを図る方法を提案する。まず、まばたき言語を制定する。次に、まばたき言語を認識するシステムを開発する。

本研究は次のような意義を有すると考えられる。① 筋組織の中でも残存性の高いといわれているまばたきを利用するため、ALS のような重篤な筋萎縮性障害者でも、専用のシステムを介せずに意思疎通を図ることができる。② 認識システムの開発により、障害者の知的活動を補助し、積極的な社会活動への参加を促すことができる。

本論文では、まず、モールス信号をベースとし、アルファベットの使用頻度に応じて瞬きパターンを割り当てたまばたき対応表を作成した。実際に作成したまばたき対応表を用いて人

平成 20 年 12 月 15 日受理

* 大学院電子電気・情報工学専攻・研究生

** 大学院電子電気・情報工学専攻・教授

対人で対話実験を行い、まばたきによるコミュニケーションが可能となることを確認した。ただし、まばたき間の時間間隔を正確にとり文字の切り替えを相手に伝える必要があることがわかった。

次に、カメラで顔画像を撮影し、発信されたまばたき言語を認識するシステムを開発した。その結果、まばたき言語の認識が可能であることを確認した。ただし、現状では、処理に時間がかかるため、処理速度を向上させる必要がある。

2. まばたき言語対応表の制定

2.1 まばたき言語とは

人間の生理現象であるまばたきに着目し、まぶたの開閉とその時間の長短を利用し、文字や記号を表現する言語である。

まぶたを閉じた時間の長短2パターンと、両目を閉じた状態・左目のみ閉じた状態・右目のみ閉じた状態の3パターン、計6パターンを組み合わせることで、さまざまな文字・記号を表現することが可能となる。

2.2 まばたき言語対応表

まばたき符合の決め方は種々考えられるが、本研究では、国際的な通用を意図して、全世界で用いられているモールス信号をベースとして制定する。

2.2.1 モールス信号とは

モールス信号とは、短点(・)と長点(-)を組み合わせ、アルファベット・数字・記号などを表現する符号方式(表1)のことで、一般に、モールス符号の短点を「トン」、長点を「ツー」と表現することが多いため、「トンツー」とも呼ばれる。

符号化方式は、標準的な英文におけるアルファベットの出現頻度に応じて符号化されており、よく出現する文字ほど短い符号で表示され、逆に使用頻度が少ないと思われる文字は長い符

表1 モールス信号表

文字	符号	文字	符号	文字	符号
A	・-	J	・----	S	···
B	-···	K	-·-	T	-
C	-·-·	L	·-·-	U	··-
D	-··	M	--	V	···-
E	·	N	-·	W	·-·-
F	··-·	O	---	X	-··-
G	-·-	P	·-·-	Y	-·-·-
H	····	Q	-·-·-	Z	-···
I	··	R	·-·		

号で制定されている。

短点・長点の組み合わせだけで構成される単純な符号のため、習得が比較的容易である。

2.2.2 まばたき言語対応表

まばたき言語対応表は、モールス信号をベースにして制定する。障害者や入院患者を対象と設定しているため、緊急時にSOS信号を用いることができるようになった。S・Oに短く簡単なパターンを割り当てる。また、アルファベットの使用頻度に応じてパターンを割り当てることで、まばたきによる疲労を軽減する。アルファベットの使用頻度は日本語は広辞苑を参考にし、英語はモールス信号をベースにウィズダム和英辞典でのアルファベット数を参考に求めてきた。それらの総合的な使用頻度を基準にしてまばたき言語対応表を作成する。以下にアルファベットの使用頻度についてまとめたものを示す(表2、および、表3)。

表3のアルファベット使用頻度表とまぶたの開閉のしやすさを考慮し、まばたき言語対応表

表2 日・英のアルファベット使用頻度表

	A	B	C	D	E	F	G	H	I	J	K	L	M	●使用頻度 ◎ 非常に多い ○ 多い △ 少ない × 非常に少ない
日本語	◎	○	×	◎	◎	×	○	◎	◎	×	○	×	△	
英語	○	△	○	◎	△	○	△	○	×	×	△	○		
	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
日本語	△	◎	△	×	△	○	◎	×	△	×	◎	◎		
英語	○	△	○	×	○	◎	△	×	△	×	△	×		

表3 総合的なアルファベット使用頻度表

	A	B	C	D	E	F	G	H	I	J	K	L	M	●使用頻度 ◎非常に多い ○多い △少ない ×非常に少ない
総合的な使用頻度	◎	△	△	△	○	×	○	△	◎	×	△	×	△	
	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
総合的な使用頻度	△	○	△	×	△	○	◎	○	×	△	×	○	△	

表4 まばたき言語対応表とモールス信号表の比較

A	■ ■ ■ ■	K	■ ■ ■ ■ ■ ■	U	■ ■ ■ ■ ■ ■
B	■ ■ ■ ■ ■ ■	L	■ ■ ■ ■ ■ ■ ■ ■	V	■ ■ ■ ■ ■ ■ ■ ■
C	■ ■ ■ ■ ■ ■ ■ ■	M	■ ■ ■ ■ ■ ■ ■ ■ ■ ■	W	■ ■ ■ ■ ■ ■ ■ ■ ■ ■
D	■ ■ ■ ■ ■ ■ ■ ■	N	■ ■ ■ ■ ■ ■ ■ ■ ■ ■	X	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■
E	■ ■ ■ ■ ■ ■ ■ ■ ■ ■	O	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Y	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■
F	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	P	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Z	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■
G	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Q	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■		
H	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	R	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	■短 ■長	
I	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	S	■ ■	■右目閉	
J	■ ■	T	■ ■	■左目閉	
				■両目閉	
				■モールス信号	

(表4)を作成した。アルファベットの使用頻度に応じてパターンを割り当てたことで、モールス信号表と比較しても全体的にパターンが短くなった。パターンが長いものでも、使用頻度が低いものを割り当てているため、まばたきの回数を減らすことができ、目の疲労を軽減することができる。また、S・Oのパターンを短くしたことで、SOSを3パターンで表現でき、緊急時に対応しやすくなった。

2.2.3 特殊符号の制定

まばたき言語対応表の制定により、身動きの取れない障害者や入院患者等と対話することは可能となった。しかし、日常的な会話全てをまばたき言語を用いて行うことは、まばたきの影響から発生する目の疲労などを考慮しても現実的であるとはいえない。そこで、日常的によく用いられるものを特殊符号として設定することで、より容易に対話を行えるようにした。

通常のまばたき言語と特殊符号の切り替え

表5 特殊符号表

■ ■ ■ ■	特殊符号への切り替え	
+		
■ ■ ■ ■ ■ ■ ■ ■	こんにちは	
■ ■ ■ ■ ■ ■ ■ ■	さようなら	
■ ■ ■ ■ ■ ■ ■ ■	眠りたい	
■ ■ ■ ■ ■ ■ ■ ■	気分が悪い	
■ ■ ■ ■ ■ ■ ■ ■	お腹が空いた	
■ ■ ■ ■ ■ ■ ■ ■	テレビが見たい	
■ ■ ■ ■ ■ ■ ■ ■	トイレ	
■ ■ ■ ■ ■ ■ ■ ■	まぶしい	
■ ■ ■ ■ ■ ■ ■ ■	暑い	
■ ■ ■ ■ ■ ■ ■ ■	寒い	
■ ■ ■ ■ ■ ■ ■ ■	騒々しい	
■ ■ ■ ■ ■ ■ ■ ■		■短 ■長
■ ■ ■ ■ ■ ■ ■ ■		■右目閉
■ ■ ■ ■ ■ ■ ■ ■		■左目閉
■ ■ ■ ■ ■ ■ ■ ■		■両目閉

は、短いまばたきを3回行うことで行う。短いまばたきを3回行った後に行ったまばたきは特殊符号のまばたきとして処理し、まばたき言語対応表とは別に特殊符号表を参照し判断する。

特殊符号の流れと制定した特殊符号表を表5に示す。

3. まばたき言語認識システムの構成

本章では、プログラムの構成、考え方などについて述べる。

3.1 全体構成

まばたき言語認識システムのハードウェア構成を、図1に示す。

・開発環境: Microsoft Visual Basic 6.0

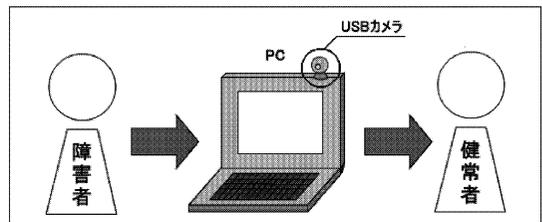


図1 まばたき言語認識システムのハードウェア構成

・ USB カメラ : USB-CAMCHAT 2
プログラムは以下の処理を行う。

- 動画のキャプチャ
- 画像の処理
 - ・ 肌色抽出
 - ・ ラベリング処理
 - ・ 中心点探索
 - ・ 目抽出
 - ・ まばたき検出
- まばたき言語対応表との照合
- 日本語変換
- 結果を出力

3.2 動画のキャプチャ

タイマー機能を用いて、USB カメラで撮影した動画を 0.2 秒間隔毎にキャプチャし保存する。これは、生理的なまばたきの間隔が約 0.1 秒なため、撮影間隔を 0.2 秒毎にすることで、生理的なまばたきを検出しにくくするためである。また、動画のキャプチャはタイマー機能の ON・OFF を切り替えることにより行う。ただし、今回は Timer 機能の ON・OFF は手動にて切り替える。その際、ファイル名に連番をつけ保存することで、時系列ごとに画像を処理することができるようにする。

このプログラムは画像処理によるまばたき検出の処理と並行して行う。動画のキャプチャの全体構成を図 2 に示す。

また、動画表示とキャプチャ処理について記す。動画表示は USB-CAMCHAT 付属のソフトを用いて行う。表示された動画をピクチャーボックスに表示し、キャプチャを行う。キャプチャ処理の流れは以下になる。

1. 初期設定
2. USB カメラで撮影した画像を表示
3. 画像の各ピクセル情報を取得
4. 作業用イメージを読み込み
5. 取得したピクセル情報を作業用イメージに貼り付ける
6. 取得したイメージをピクチャーに表示

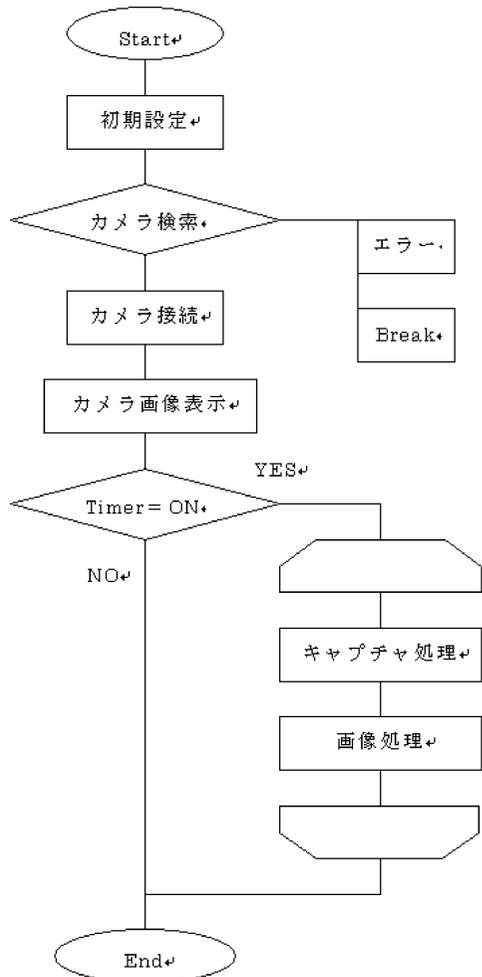


図 2 動画キャプチャの全体構成

7. ファイル名に No を付けて画像を保存
キャプチャする際に一旦作業用イメージに画像情報を貼り付け、その後、ピクチャーボックスに表示する。この処理をすることで画像の保存・編集を行うことが可能となる。また、この画像を使用し以後の処理を行う。

3.3 まばたき検出プログラム

3.2 で保存した画像を用いて、画像処理を行いまばたきを検出する。画像の処理の手順は肌色抽出・ラベリング処理・中心点探索・目元抽出・

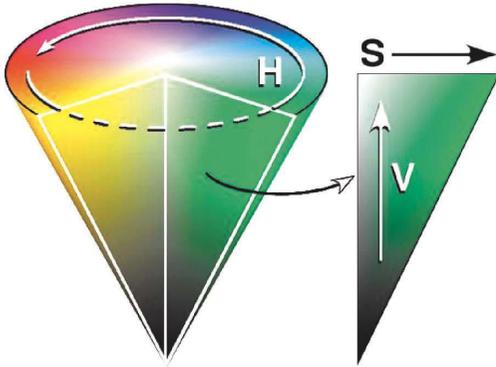


図3 HSV表色系

まばたき検出の流れで行い、以下にそれぞれの処理について記す。

また、画像の処理が終わった画像は順次削除することで、ハードディスクの容量を軽減する。

3.3.1 肌色抽出

撮影した画像から目元抽出に必要な顔部分の抽出を行うため、画像内から肌色の範囲を抽出し不要な背景部分を除去する。この肌色範囲の抽出はHSV表色系を用いて処理を行う。

HSV表色系とは、色相(Hue)・彩度(Saturation)・明度(Value)の3つの成分からなる色空間のこと(図3)で、RGB表色系と違い、色相・彩度・明度の各要素を独立して処理を行えるため、肌色抽出に適している。

肌色の抽出は以下の手順で行う。

- ① 3.2で保存した画像を読み込む。
- ② ピクセル単位で色情報を取得する。ただし、この時点ではRGB表色系で色情報が取得される。

$$\begin{aligned}
 &MAX = \max(R, G, B) \rightarrow V = 0 \text{ 以外の場合} \\
 &MIN = \min(R, G, B) \\
 &V = MAX \\
 &V = 0 \text{ の場合} \\
 &S = 0 \\
 &H = 0 \\
 &S = (MAX - MIN) / MAX \\
 &Cr = (MAX - R) / (MAX - MIN) \\
 &Cg = (MAX - G) / (MAX - MIN) \\
 &Cb = (MAX - B) / (MAX - MIN) \\
 &H = \begin{cases} Cb - Cg & R = MAX \\ 2 + Cr - Cb & G = MAX \\ 4 + Cg - Cr & B = MAX \end{cases} \\
 &H = 60 \times H \\
 &H = H + 360 \quad H < 0
 \end{aligned}$$

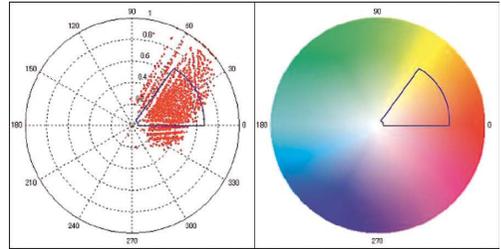


図4 全学生のプロット図とHSV表色系での色範囲

- ③ 取得したRGB値を左記の変換式を用いてHSV表色系に変換する。
- ④ 肌色範囲以外の不要な部分を除去し、顔範囲を抽出する。

また、肌色の範囲指定は以下の条件式を用いて行う。

$$H = 0 \sim 53 : S = 0.05 \sim 0.65 : V = 0.4 \sim 1$$

この条件式は、八戸工業大学システム情報工学科の全学生(約400名)の肌色データを抽出し、その平均値を求めた値である。

肌色データのプロット図とHSV表色系での色範囲を図4に示す。

3.3.2 ラベリング処理

ラベリング処理とは、隣接する色を一つのまとまりとして扱うための識別表を貼り付ける処理である。抽出した肌色範囲にラベリング処理を行い、最大ラベルのみを残すことで、肌色抽出で除去できなかった背景部分などを除去し、処理に必要な顔部分を抽出することができる(図5)。以下にラベリング処理の手順を示す。

- ① 中心から左上へ向かって捜査を行う。
- ② 色部分を見つけた場合ラベルが付加され

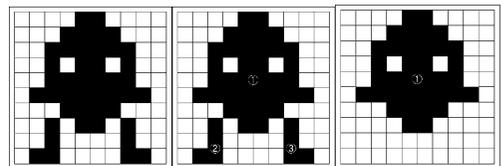


図5 ラベリング処理の流れ



図6 元画像

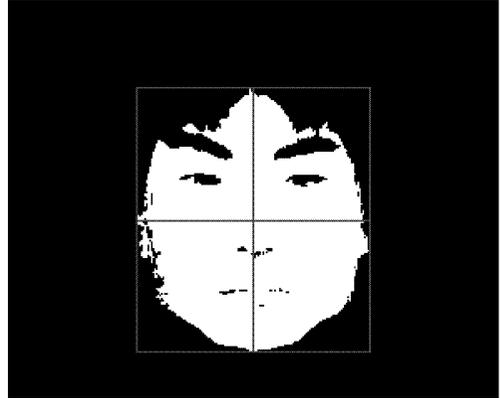


図8 中心点探索



図7 ラベリング処理

元画像と肌色抽出・ラベリング処理を行った結果を図6・図7に示す。

まず、元画像に肌色抽出を行い、処理に必要な顔部分の抽出を行う。ただし、この時点では肌色に近い色範囲の背景や、不要な肌色部分がノイズとして残っている。そこで、その画像にラベリング処理を行うことで、不要な背景などのノイズを除去し、処理に必要な顔部分を正確に抽出することができる。

3.3.3 中心点探索

ラベリング処理を行い抽出した顔部分の中心点を探索し、以降の処理で基準として用いるための処理。同時に、顔部分の範囲を取得することで、不要な走査処理を減らし処理速度の向上を図る(図8)。中心点探索の処理手順について以下に示す。

- ① 左上から右下へ向け右方向へ向かい走査を行う。
- ② 走査を行い最初に見つかったラベル部分を顔の上端とする。
- ③ 同様の処理を行い、顔の右・左・下端を探索する。
- ④ 取得した端点の座標から顔部分の範囲と、その中心点の座標を取得し、以後の画像の基準として用いる。

3.3.4 目元抽出

目とは顔の中心よりも必ず上にある。よって、

ているかを確認する。

- ・ラベルが付加されていない場合
新たなラベルを付加し、上下左右の探索を行う。
探索を行い隣接部分に肌色範囲があった場合、現在付加されているラベルを隣接部分にも貼り付ける。
 - ・ラベルが付加されている場合
上下左右の探索を行い、隣接部分に肌色範囲があった場合、現在付加されているラベルを隣接部分にも貼り付ける。
- ③ ②の処理を右上・左下・右下にも行う。
 - ④ 最大ラベルのみを残し、不要なラベルを除去する。

その特性を利用し顔部分の中心より上方の探索を行うことで目元を抽出する。前処理で取得した中心点と顔部分の範囲を基準にし、目元の抽出を行う。目元抽出の手順を以下に示す。

- ① 左目を抽出するために、前処理で取得した中心点の座標より右上を対象とし、処理を行う。
- ② 中心点から右上へ、右方向へ向かって走査を行う。この際、目の位置が右端によることはないという考え方を元に、顔部分の4分の3の範囲で走査することで、処理速度を向上する。
- ③ 走査処理を行い、初めに見つかったラベル部分を目の下端とする。
- ④ さらに走査処理を続け、ラベル部分が見つからない部分が現れるまで走査処理を続ける。
- ⑤ ラベル部分が見つからない部分が現れたら、その前の段を目の上端とする。
- ⑥ 次に左端を探索するために、中心点から右上へ上方向に向かい走査を行い、目の左端の座標を取得する。この際、走査する範囲を取得した上端座標までにすることで、処理速度の向上を図る。
- ⑦ ④、⑤と同様の処理を行い、目の右端の座標を取得する。
- ⑧ 取得した上下左右の端点座標から、左目

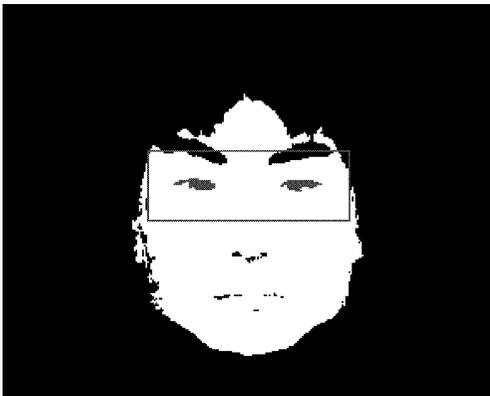


図9 目元抽出処理

を抽出する。

- ⑨ 同様の処理を左部分にも行い右目を抽出し、目元の抽出を行う(図9)。この際、目元の領域を取得し以後の処理で用いる。また、両目間の中心点も同時に取得し、まばたきの検出に用いる。

3.3.5 まばたき検出

目を閉じた状態と目を開いた状態を比較するとき、目を閉じた状態の方が目を開いた状態よりも目元領域の範囲は狭くなる。その考えに基づき、目元領域の変化量からまばたきの状態を検出する。以下にまばたき検出の手順を示す。

- ① 次の時系列の画像を読み込み、肌色抽出から目元抽出までの処理を行い、目元領域を取得する。この際、前画像の処理を行ったときに取得した、両目間の中心点を基準に範囲を狭めて処理することで、処理速度の向上を図る。
- ② 現時系列で取得した目元領域と前時系列で取得した目元領域の変化量を比較する。

・変化量が40%以下の場合

まぶたが開いた状態(図10)とし、
右目が開いている場合

Pattern = +1

左目が開いている場合

Pattern = +2

・変化量が40%以上の場合

まぶたが閉じた状態(図11)とし、
右目・左目共に Pattern = +0

- ③ 取得したPatternの値を基準にして、まばたきの状態を検出する。

・Pattern = 0の場合

両目が閉じた状態

・Pattern = 1の場合

右目開・左目閉の状態

・Pattern = 2の場合

右目閉・左目開の状態

・Pattern = 3の場合

両目が開いた状態



図10 まぶた開



図11 まぶた閉

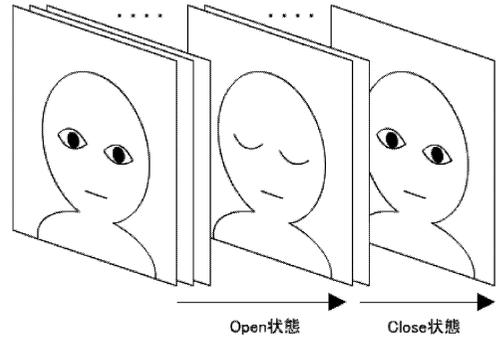


図12 まばたき間隔のイメージ閉

3.4 まばたき言語対応表との照合

一定間隔(0.2sec)毎に撮影した画像に上記の処理を行い、まばたき Pattern をそれぞれ取得する。その際、Pattern が変化するまで Count 値を加算していき、Pattern の値が変化した場合、まばたきの状態が変化したものとする。それまでに加算された Count 値から文字の切り替えを判断し、対応表との照合を行う。

まぶたが開いている状態から閉じた状態になるまでを Open 状態とし、閉じた状態から開けるまでを Close 状態とする(図12)。Close 状態の間の Count 値からまばたきパターンを取得し、Open 状態の Count 値から文字の切り替えを判断する。

- ① Close 状態の Count 値
 - ・Count=1: 生理的なまばたきとして認識するため除外する。
 - ・Count=2~4: 短いパターンのまばたき
 - ・Count=5~: 長いパターンのまばたき
- ② Open 状態の Count 値
 - ・Count=1~9: 同一の文字として認識する。
 - ・Count=10~: 以後のまばたきは別の文字として処理するため、今までに取得したパターンから文字を認識する。
- ③ 取得したパターンとまばたき言語対応表を照合し、該当する文字を表示する。

3.5 日本語変換

これまでの処理でまばたきをアルファベットで画面上に表示することができた。ただし、アルファベットそのままでは、相手にわかりづらいため日本語に変換する必要がある。そこで、ローマ字入力によりアルファベットを日本語に変換する方法と、携帯電話の入力方式を参考にしたもの、二つの変換方法を考案した。

① ローマ字変換

- ・入力例) ゆめ……Y・U・M・E
- ・メリット

アルファベットをそのまま使用し変換するため、英語圏でも用いることができる。

- ・デメリット

ローマ字全26種を覚える必要があり、習得するまでに時間がかかる。

② 携帯電話の入力方式

- ・入力例) ゆめ……82・74
- ・メリット

携帯電話の普及によりどの世代にも受け入れられやすい。特に若い世代に受け入れやすく、覚える種類も数字10種でよいから、習得しやすい。

- ・デメリット

覚えるのが数字のみであり、50音の日本語に対応させているため国内のみでし

か使えない。そのため海外では使えないため国際的ではない。

本研究では、覚える数は多くなるものの、緊急時に用いる SOS 信号との対応や普及のしやすさ、また、国際社会でも意思疎通が図れる点も考慮しローマ字変換方式を採用した。

4. まばたき対話実験の実行結果と考察

制定したまばたき言語を用いて、実際に人対人での対話および、まばたき言語認識プログラムを用いた人対機械で対話実験を行った。

・人対人

制定したまばたき言語を用いて、本研究室の学生2名と実際に対話実験を行った。実験は以下の手順で行った。

- ① 「あ」～「お」までの母音5文字から1文字
- ② 「あ」～「ん」までの母音+子音の組み合わせから1文字
- ③ ②+濁点・半濁点を含む全文字から1文字
- ④ 「おはよう」等の簡単な単語
- ⑤ 特殊符号の読み取り

対話実験の結果、①の母音1文字単位での対話と⑤の特殊符号の読み取りは成功した。しかし、②の母音と子音を組み合わせた場合では読み取れないことが多く、③、④ではほぼ読み取れなかった。

①、⑤が成功し、②～④がうまく読み取れなかった原因は、文字の切り替えをするための間を十分に取らなかったことにあると思われる。対応表を見て分かるように、母音は全てまばたきのパターンが1つで終わるため、間を取る必要はない。しかし、②～④ではアルファベットを組み合わせる必要があるため、まばたきのパターンは2以上になり文字と文字の間に間を取る必要がある。そのため、文字と文字を切り替える間隔を取る必要のない⑤の対話は成功

し、同一文字内のパターン変更の間隔と文字切り替えの時間間隔を取る必要のある②～④の対話は失敗したものと考えられる。よって、同一文字内のパターン変更の間隔なのか、文字と文字を切り替えるための間隔なのかを、正確にわかるように相手に伝える必要がある。

また、対話をするために必要なアルファベットのまばたきパターンを習熟していなかったため、どのまばたきパターンなのかを正確に判別できなかったことも原因だと考えられる。

・人対機械

静止画像からのまばたき検出・タイマー機能を用いた動画キャプチャの処理は完成したが、時系列画像からのまばたき状態の変化を検出するプログラムと組み合わせないため、動画像処理には対応することができていない。

5. 結 論

本研究では、まばたき言語対応表を制定し、それを用いて対話実験を行った。また、まばたき言語認識システムのプログラムを設計・作成し、静止画像からのまばたき検出可能なプログラムを作成した。

モールス信号をベースとし、アルファベットの使用頻度に応じて瞬きパターンを割り当てたまばたき対応表を制定した。制定したまばたき対応表を用いて実際に対話実験を行い、まばたきによるコミュニケーションが可能となることを確認した。また、カメラで顔画像を撮影し、発信されたまばたき言語を認識するシステムを開発し、まばたき言語の認識が可能であることを確認した。

ただし、対話実験により、まばたき間の時間間隔を正確にとり、文字の切り替えを相手に伝える必要があることがわかった。

また、現状では、光源・髪型等の影響により、

正確に目元の抽出を行えない場合もある。また、画像の処理に時間がかかるため、処理速度を向上させ、動画像を用いたりリアルタイム処理に対応させる必要がある。

参考文献

- [1] 古瀬, 渡辺, 二見, 星宮, 半田, “運動機能麻痺者のための残存運動機能を用いた制御命令入力システム”, 医用電子と生体工学, Vol. 37-2, pp. 152-160, (1999-)
- [2] 橋本, 米澤, 伊東, “ポインティングデバイスのスイッチ代行機構への歯の接触音とハミングの応用”, 電子情報通信学会論文誌 D-I, Vol. J84-D-I, No. 7, pp. 1112-1118, (2001-7)
- [3] 古川勇樹, 苫米地宣裕, “まばたきによるコミュニケーションの試み”, 平成 18 年度情報処理学会東北支部研究会 (弘前大学), No. 10, (2007-2)
- [4] 古川勇樹, 苫米地宣裕, “まばたき言語認識システム”, 平成 19 年度情報処理学会東北支部研究会 (八戸工業大学), No. 11, (2007-12)