## 法政大学学術機関リポジトリ

#### HOSEI UNIVERSITY REPOSITORY

PDF issue: 2025-07-04

複数の点滅視覚刺激環境下における定常状態 視覚誘発電位(SSVEP)による脳波の識別に関 する研究

松尾, 拓実 / MATSUO, Takumi

(開始ページ / Start Page)

1

(終了ページ / End Page)

59

(発行年 / Year)

2016-03-24

(学位授与年月日 / Date of Granted)

2016-03-24

(学位名 / Degree Name)
修士(工学)

(学位授与機関 / Degree Grantor)

法政大学 (Hosei University)

# 2015 年度 修士論文

論文題目	複数の点滅視覚刺激環境下における
定常状態視覚	誘発電位(SSVEP)による脳波の識別に関する研究
<sup>ふりがな</sup> 氏名	まっぉ たくみ 松尾 拓実
研究科専攻	理工学研究科機械工学専攻
学籍番号	14R1134
指導教員	石 井 千 春
修了年月(西暦)	2016年 3月
	法政大学大学院

## Abstract

When the disease called Amyotrophic Lateral Sclerosis (ALS) had progressed, the muscular strength of patient decreases. Then, patients cannot control a wheelchair by their own ability. In this case, Brain Machine Interface (BMI) helps such patients to control the electric wheelchair by their own intention. BMI enables a human to control a machine using the electroencephalogram (EEG).

In this study, a detection method of steady state visual evoked potential (SSVEP) in short time was investigated for the sake of future use in BMI. In addition, experiments for distinguishing SSVEP were examined under the condition that plural visual stimuli were presented simultaneously.

Preliminary experiments were carried out to determine the form and color of a blinking symbol and the number of the electrodes to be used. As results, the blinking symbol of 10Hz and 12Hz was chosen as white square with black background, and the blinking symbol of 15Hz was chosen as white square with blue background. In addition, three electrodes were chosen to be used. Using the short time Fourier transform (STFT) and the moving-average method, the analysis method of electroencephalogram (EEG) for detecting SSVEP and detection conditions of SSVEP were determined.

Then, experiments for distinguishing SSVEP were executed under the condition that three kinds of the blinking animation, in which the white square symbol blinks in 10Hz, 12Hz, and 15Hz respectively, were presented simultaneously as visual stimulus. Accuracy rate of more than 90% was obtained for distinction of each stimulus frequency in 2 to 6 sec.

## 目次

第1章	緒論
1-1.	研究背景
1-2.	問題定義
1-3.	研究目標
1-4.	本稿の構成
第2章	実験環境
2-1.	刺激提示用ノート PC5 -
2-2.	視覚刺激提示ソフトウェア 6 -
2-3.	AviUtl によって作成した視覚刺激用動画
2-4.	脳波計7 - 7 -
2-5.	MATLAB/Simulink - 9
第3章	脳波 10 -
3-1.	脳波とは 10 -
3-2.	自発脳波と誘発脳波 11 -
3-3.	BMI の研究例 11 -
3-4.	定常状態視覚誘発電位(SSVEP) 13 -
3-5.	脳波の測定位置 14 -
第4章	予備実験
4-1.	本研究で用いた解析手法15 -
4-2.	動画の色の組合せを決める実験15 -
4-3.	図形の形を決める実験17 -
4-4.	SSVEP を検出するための周波数を調べる実験 19 -
4-5.	電極を減らすための実験20-
第5章	SSVEP の検出実験
5-1.	SSVEP の検出条件
5-2.	SSVEP の検出実験結果25 -
5-3.	考察26 -
第6章	複数の光刺激提示下での SSVEP の識別
6-1.	2 種の点滅動画による SSVEP の識別実験
6-2.	2 種の SSVEP の識別条件 28 -
6-3.	2 種の点滅動画による SSVEP の識別実験結果29 -
6-4.	3 種の点滅動画による SSVEP の識別実験 32 -
6-5.	3 種の点滅動画による SSVEP の識別実験結果 33 -
6-6.	考察35-

第7章	3種の点滅動画によるランダム注視実験	36 -
7-1.	3種の点滅動画によるランダム注視実験	36 -
7-2.	3種の点滅動画によるランダム注視実験結果	37 -
7-3.	考察	38 -
第8章	結論	39 -
謝辞		40 -
参考文献	t	41 -
付録		42 -
A)	使用機器使用	42 -
A-1)	)視覚刺激提示用ノート PC	42 -
A-2)	) 脳波計測機器	43 -
B)	ソフトウェア	44 -
B-1)	) C++言語プログラム	44 -
B-2)	) MATLAB/Simulink モデルプログラム	51 -
B-3)	) MATLAB M 言語プログラム	51 -
B-4)	) MS-Excel Visual Basic プログラム	54 -

#### 第1章 緒論

本章では、研究背景と問題定義、目標について述べ、最後に本稿の構成について示した.

#### 1-1. 研究背景

身体機能に障がいを持つ患者が介護者に頼ることなく自由に行動するために、杖や歩行器、車椅子など様々な福祉機器が開発されてきた.これらの福祉機器を使用するには他の身体機能を使用することによって代替、もしくは補助する場合が多くある.例えば車椅子の場合では、脚の代替として腕を用いて車輪を回し、移動を可能にしている.

しかし身体機能が代替できない程度に低下している場合や、欠損などによって喪失している場合では、これらの福祉機器を使用できない。例として、筋萎縮性側索硬化症(ALS)や筋ジストロフィー、四肢麻痺を挙げる。

ALS[1]とは、筋肉を動かし、かつ運動をつかさどる運動ニューロンだけが障害をうけ、その結果、脳から「手足を動かす」などの筋肉を動かす命令が伝達できなくなり、筋力が低下していく病気である。また筋ジストロフィー[2]とは骨格筋に発現する遺伝子の変異・発現調節異常により、蛋白の喪失・機能異常が生じ、筋細胞の正常な機能が破綻して変性・壊死に至るため、その結果、運動機能低下が引き起こされる。四肢麻痺[3]は、事故によって頸髄や脳などの外的要因で起こるものと遺伝性の病気で起こるものがあり、両肩・両腕などに向かう神経が機能しなくなり、筋肉の弛緩が起きる症例である。

そこで、そのような病気を抱える患者でも使用できるシステムとして脳の信号を用いるブレイン・マシン・インタフェース (BMI)が挙げられる。BMI とは、人と機械との間で意思や情報の仲介のために脳の信号である脳波を解析して電気信号の形で出入力するためのプログラムや機器である。BMI を用いることによって、電極で計測した脳波の情報を機器に出力することが可能になる。BMI の例として、Fig.1-1 に示したフランス国立科学研究センター (CNRS) と産業技術総合研究所の共同ラボ「AIST-CNRS ロボット工学連携研究体」で行われているヒューマノイドを動かすためのBMI[4][5]を紹介する。このBMI は、ロボットをアバターとして使用することで身体的機能障害を持つ人の運動機能を実現し、物理的なやり取りを可能としている。





a) Patient side

b) Humanoid side

Figure 1-1 Example of the BMI

#### 1-2. 問題定義

ブレイン・マシン・インタフェース (BMI) は、脳情報を読み取る方法に侵襲式と非侵襲式の2種類がある. 侵襲式のBMIでは、脳に直接多数の細い電極を取り付けることによって脳情報を読み取る. 空間的・時間的分解能も極めて高く、ほぼリアルタイムで検出できるが、電極を埋め込むための手術が必要であること、電極の通信線を露出させている部分に起こる感染症などのリスクがある.

非侵襲式では、頭皮上あるいは頭部近くから脳活動を検出するため、侵襲式で必要となる手術や感染症のリスクはない、非侵襲式はさらに機能的磁気共鳴画像(fMRI)と脳波測定の2種に分かれる。fMRIとは、神経活動がより大きく生じている脳部位を、血流中の還元へモグロビンの変化から推定する方法であるが、大掛かりな装置の中に仰臥して入る必要があるため、現状 BMI として実用化するのは難しい。

脳波測定とは、頭皮上に装着した電極から脳活動を電気信号として測定する方法である. 測定技術もあまり困難でないため、臨床から研究まで広く使われる方法である.

脳波測定を用いた BMI の研究では、BMI を通じて機器に行わせたい操作や機器の操作に対応した方向などの概念、身体運動の想起を行い、その時の脳波を読み取って BMI を動作させる被験者の意識の変化を利用する方法と、機器の操作に対応した外部刺激を受けた時の脳波を読み取って BMI を動作させる被験者の外的刺激に対する反応を利用する方法がある.

前者では、被験者が機器の操作内容、操作に対応した方向などの概念を想起した時の脳波を周波数解析や波形自体の特徴を抽出する解析を行い、解析結果を基準データや閾値と照らし合わせる場合と、解析結果を学習モデルなどに繰り返し学習させ、想起内容ごとに分別させる場合の研究が一般的である。しかしながら、運動想起や方向などの概念の想起は、脳波からの意識の検出には個人差が大きく、また必ずしも検出できるとは限らない。また概念の想起では、頭部のどの部位に表れるか調べるために電極で頭部を覆い尽くすため、電極数が多くなり、また学習モデルに学習させる方法では、十分な学習を行うための回数が必要であるため、被験者に負担となる場合がある。

後者では、被験者に光刺激や音刺激などの外部刺激を与え、その時の脳波を前者と同様に周波数解析や波形の特徴抽出解析、解析結果の学習などを行う研究が一般的である. 概念の想起とは違い、外部刺激を受け取る感覚の脳における処理領域が大まかに分かっており、特に光刺激を受け取る視覚は後頭部に広く分布されていることが知られており、光刺激による脳波を利用した BMI の研究もある. しかし、これらの研究では LED などの輝度が高い光源を使用している場合が多く、光刺激が強すぎると、光てんかんを引き起こす可能性がある. 光てんかんとは、光に過敏な体質である患者が点滅に似た光刺激を受け取ると、感受性が高い患者ほどてんかんに似た意識喪失や筋痙攣を引き起こす. 光に過敏でなくとも、気分が悪くなることがある.

#### 1-3. 研究目標

本研究では、将来的に電動車椅子を BMI により制御するための基礎研究として、BMI に使用できる脳波の解析手法を模索することを目標とする。今回は、オンラインでの運用に備えて、点滅する光刺激から一般的に安定して表れる、定常状態視覚誘発電位(SSVEP)を短時間で検出する方法を検討した。光てんかんが起きないようにするため、光刺激が弱いと考えられる動画を用いて PC 画面上で視覚刺激を与え、被験者の負担とならないように極力電極数を少なくした条件において、学習アルゴリズムを必要としない検出方法を考えた。そしてこの検出方法を用いることで、複数の視覚刺激が与えられている状態で、注目した視覚刺激の影響による SSVEP を識別できるかどうかの検証を行った。なお、本研究ではすべての実験に対して、健康な 24 歳の右利きの成人男性 1 人を被験者とした。

#### 1-4. 本稿の構成

以上説明した研究背景、問題定義、研究目標から本稿の章立てを以下のようにして構成した.

第2章では本研究を行うための装置を始めとした実験環境について述べる。第3章では本研究で測定する生体信号である脳波について測定方法と、本研究で利用する SSVEP の特徴について述べる。第4章では光刺激として用いる動画の設定や、ある周波数で図形が点滅する動画を用いた時にどの周波数の SSVEP で表れるか、電極配置や数の設定などを決定するための予備実験について述べる。第5章では、予備実験によって定めた動画を用いて、考えた SSVEP の検出方法で単一の視覚刺激による SSVEP が検出できるかの実験、その実験の結果、結果を受けての考察を述べる。第6章では、複数の光刺激を提示した状態で注目した動画による SSVEP の識別実験、実験の結果、考察を述べる。第7章では、それぞれの動画の配置を固定した、複数の光刺激を提示した状態でランダムに注目した動画による SSVEP の識別実験、実験の結果、考察を述べる。第8章では前章までの研究成果から本研究の結論と今後の展望について述べる。

### 第2章 実験環境

本研究では、Lenovo 社製のノート PC である ThinkPad X240 を刺激提示用光源とし、C++ 言語によって作成した視覚刺激提示ソフトウェアや無料の動画編集ソフトウェアである AviUtl を用いて作成した視覚刺激用動画を用いて刺激を与え、株式会社デジテックス研究 所が開発した脳波計で視覚刺激を与えた時の脳波で測定し、米 The Math Works 社の製品である MATLAB/Simulink を用い、脳波解析に用いるソフトウェアによって解析した。本章は実験及び解析に用いた装置及びソフトウェアについて述べる.

### 2-1. 刺激提示用ノート PC

本研究では、視覚刺激の光源として光てんかんを起こしづらいと考えられる輝度を持つノート PC である Lenovo 社製の ThinkPad X240 (Fig.2-1) を用いる. ノート PC を選んだ理由として、電動車椅子を BMI により制御するための基礎研究であるため、モニターと筐体が一緒なためコンパクトであるという利点と、一般的に販売されている製品を使用することによって特別な機器が必要ないという利点による. 詳細は付録に述べる.



Figure 2-1 Appearance of ThinkPad X240

#### 2-2. 視覚刺激提示ソフトウェア

点滅する図形の色と背景色により SSVEP がより強く表れる組合せを調べるための予備 実験のために C++言語によって Fig.2-2 に示したようなソフトウェアを作成した. 作成した プログラムでは,何回点滅させるかを決定する出力回数,点滅の時間間隔を決定する出力時間(単位:ms)を入力し,「ファイルを点滅」ボタンを押下することで,中央の図形の表示,非表示を繰り返し,点滅を表現する. 図形の色と背景色の変更が可能である.

WindowsOS はリアルタイム OS ではないため、出力時間にばらつきが発生するが、並行で動作している点滅している時間間隔を確認するプログラムでは、10[ms]単位でのばらつきは見られず、また実験に用いる周波数の範囲で視覚的に確認を行ったが、点滅間隔のばらつきがわからないため、予備実験時では問題ないと考えた.

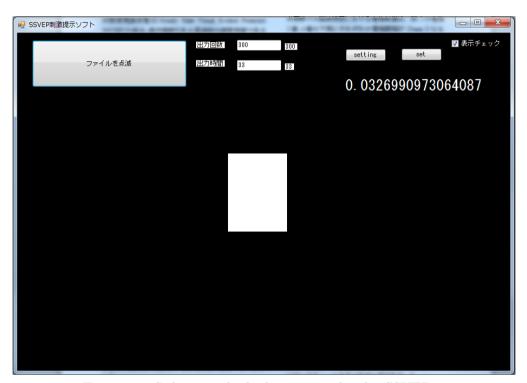


Figure 2-2 Software which shows stimulus for SSVEP

#### 2-3. AviUtlによって作成した視覚刺激用動画

AviUtl[6]とは、AVI ファイルにノイズ除去フィルタやシャープフィルタなどの各種フィルタをかけることや、動画のリサイズ、動画連結、テキストや図形などのレイヤーの重ねることが可能な動画編集ソフトウェアである.機能拡張プラグインを導入することで、MP4などの AVI ファイル以外の動画ファイルも編集でき、フィルタの追加も可能となる.また動画には、フレームレート(単位:fps)と呼ばれる動画が 1 秒間に画像が何枚で構成されているかを示す情報があり、AviUtlでは最大 60fps の動画が作成可能である.

動画作成では、背景が黒の基礎レイヤーを用意し、一定フレーム間隔毎に図形を中央に配置するレイヤーを基礎レイヤーに重ねた.この時一定フレーム間隔が点滅する視覚刺激の周波数となり、24fps、60fpsの素数の組合せから用意した視覚刺激周波数は8Hz、10Hz、12Hz、15HzのFig.2-3のような4種類の動画を用意した.

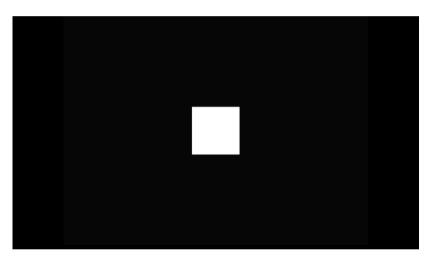


Figure 2-3 The blinking animation as visual stimulus

#### 2-4. 脳波計

本研究では被験者の脳波を測定する脳波計として株式会社デジテック研究所製の生体信号記録装置である PolymateII を用いる. (Fig.2-4-a) 生体アンプとバッテリーを内蔵した最大 12ch 生体信号の測定が可能な装置で、Fig.2-4-b に示すようなアクティブ電極を標準装備している. 主な測定可能現象は、脳波の他、眼球運動、心電図、表面筋電図、脈波、呼吸、SpO2、いびき、体位、体動、EDA、体温などの測定が可能となっている. また、各種解析ソフトで睡眠解析、事象関連電位、CDM などの解析が可能である. 最大記録サンプリング周波数は 1000Hz で、交流ノイズの影響を軽減させるため、コンパクトフラッシュカードへの記録とバッテリー駆動によるスタンドアローンでの測定が可能である. 本研究では、脳波のみの測定に用い、他の生体信号の記録は行わない.

脳波測定時には、Fig.2-4-b に示すように専用のアクティブ電極の測定金属部に脳波測定用のペーストを盛り付け、髪の毛をかき分けるようにして、頭皮上に張り付ける.電極の上からガーゼを当てることで、電極を張り付けた際にあふれてしまうペーストを接着させ、電極がずれてしまわないように、髪の毛が電極を頭皮から押し離さないように固定する.



a) PolymateII



b)Active electrodes Figure 2-4 Apparatus in EEG measurement

#### 2-5. MATLAB/Simulink

MATLAB は科学・工学分野の様々な数値計算やデータ解析、シミュレーション、可視化のための統合環境を提供する開発ソフトウェアである. プログラミング言語には、C言語に代表されるコンパイラ型と Basic 言語に代表されるインプリンタ型が存在するが、MATLAB はインプリンタ型のプログラミング言語である. インプリンタ型はコンパイラ型に比べて、命令の実行速度は遅いものの、プログラムを部分単位で実行可能なことから容易にプログラムの変更・修正を行うことが可能である.

MATLABはC言語がエンジンとなっているためインプリンタ型の言語としては比較的早い処理が可能であり、配列宣言を行うことなく行列計算を容易に行うことができるので、行列計算を多用する制御や信号処理に適応しやすい。その他の数学的算術も各種関数を用いることで容易に行うことができる。

MATLAB がプログラミング言語であるのに対し、Simulink はブロック線図を用いてシステムを構築するソフトウェアである。Simulink は MATLAB のプロダクトファミリの 1 つであり、MATLAB との正確な連携が可能である。Simulink では論理演算や算術演算、伝達関数のブロックを組み合わせることで数学モデルのシミュレーションや信号処理など様々な環境構築ができる。また、既存のブロックだけでなく、MATLAB で作成したプログラムや C言語、Fortran、BASIC言語などで作成したプログラムをブロックとして Simulink のブロック線図に加えることができる。例として、Fig.2-5 に定常状態視覚誘発電位 (SSVEP) の検出を自動で行うプログラムを示した。

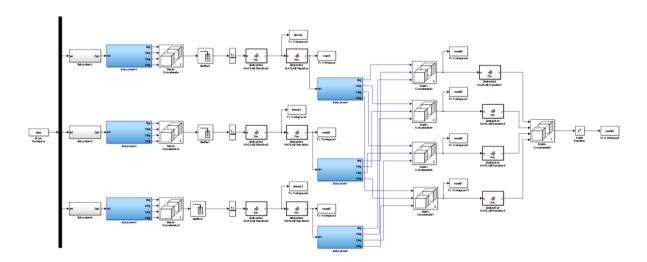


Figure 2-5 MATLAB/Simulink Program

### 第3章 脳波

本章では 2 章にて説明を行った脳波計によって測定する脳波現象やその測定位置や手法について記述する.

#### 3-1. 脳波とは

脳波(ElectroEncephaloGram: EEG)とは脳内の神経細胞の電気活動を頭皮上あるいは脳深部に設置した電極によって測定、記録したものである. Fig.3-1 は本実験で測定した脳波の原波形である. 医療の臨床検査で測定した波形を直接記録、観察することでてんかんを始めとした脳に関連する障がい、疾患の診断に用いられている.

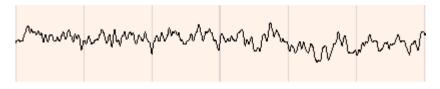


Figure 3-1 EEG

脳波は被験者の状態や感情に応じて優位となる周波数帯が異なる. その特性から,各周波数帯における脳波には名称が付けられており,それぞれ被験者の状態を示す特徴がある. その中でも覚醒時に優位的に発生する脳波及びその特徴を Table 3-1 に示す.

	Slow wave			Fast wave		
Wave Name	δ	Θ	α	Middle speed wave	β	γ
Band [Hz]	0.5-3	4-7	8-13	14-17	18-30	30-
Characteristic	Non REM Sleep	Drowsiness	Relaxed	Anxious Thinking	Busy mind activity	Excitement

Table 3-1 Characteristic of Each Brain Wave

また、被験者に刺激を与えた場合に、誘発電位等の脳波の変化を測定することができ、 その特徴から刺激に対してどのような反応をしているか、どの程度感情変化が起きている かを観測することができる。実際に音刺激に対して発生する誘発電位の有無が、脳死判定 の判定材料の1つとなっている。

#### 3-2. 自発脳波と誘発脳波

人間の脳に発生している脳波には、自ら行動や思考することによる自発的な脳波(自発 脳波)と五感から受け取る刺激に対する反応として誘発される脳波(誘発脳波)に大きく 分類される.

#### A) 自発脳波

人間の脳は常に何らかの活動をしているため、睡眠中でも、何らかの脳波が観測される.このように、外部からの刺激がなくても出ている脳波を自発脳波と呼ぶ.自発脳波の種類として、運動する若しくは運動の想起を行うことによって発生する事象関連脱同期(Event-Related Desynchronization: ERD)や閉眼時に起きるα波の増大現象がある.

#### B) 誘発脳波

人間の脳は、外界からの刺激(視覚刺激、聴覚刺激、触覚刺激等)に対し、何らかの反応を示す。この時に観測される脳波を誘発電位と呼ぶ、誘発脳波の種類として、周波数的な視覚的な刺激によって発生する定常状態視覚誘発電位(Steady State Visual Evoked Potential: SSVEP)や身体に刺激を与えることによって発生する体性感覚誘発電位(Somatosensory Evoked Potentials: SEP)がある。

#### 3-3. BMI の研究例

近年、解析技術の進歩から工学的分野での開発が著しく、多くの研究機関でブレイン・マシン・インタフェース(Brain-Machine Interface: BMI)と呼ばれる、脳波の解析結果から判断し、動作する機器の開発が進められており、実際に脳波で操作するロボットや車椅子の開発が進められている。BMIの研究例をいくつか紹介する.

Donoghue[7][8]らは、ヒトの運動野での脳波現象を利用した BMI の開発に向けた臨床試験を報告した. 四肢麻痺を患った被験者に対して、脳の一次運動野に侵襲式の 96 本の微小電極が並んだ装置(ブレインゲイト)を取り付け、運動意図に基づく運動野ニューロン集団の活動パターンをインタフェースが読み取り、機械信号に変換することで、文献[7]ではパソコンのカーソル操作やクリック、文献[8]では義手の操作を行った. Hajibabazadeh[9]らの研究では、被験者に左右の腕の運動想起を行わせ、頭皮上に取り付けた 6 個の電極で脳波を測定し、測定した脳波を Wavelet 解析によって周波数領域を行った後、その解析データから Support vector machine(SVM)というパターン認識手法によって左右の腕の運動想起の分別を行い、その認識結果に基づいてピューマ・ロボットの操作を行った. Noda[10]らの研究では、リハビリテーション・トレーニングを補助する脚部の外骨格ロボットを脳波で操作するシステムを開発した. 脳波から上、下の意識を分類し、それぞれ立ち上り、腰掛け動作を補助するように外骨格ロボットを制御した. MANO[11]らは、ライントレースにより移

動する電動車椅子が分岐点に差し掛かった時に、搭乗者の頭皮上に取り付けた 15 個の電極で脳波を測定し、搭乗者の操縦内容(前進、右旋回、左旋回)の意識を読み取ることにより、電動車椅子を操作した。また茨木[12]らは、Emotiv 社製の非侵襲式の BMI デバイス (Fig.3-2) を用いて、搭乗者の意識(前、右、左)を検知し、それぞれの意識に対応する指令(前進、右折、左折)を電動車椅子に出力するシステムを開発した。2 時間弱の訓練の後、3 種類の意識の認識率は 84~91%であり、電動車椅子の制御を高精度に行った。また Jiralerspong[13]らは、文献[12]と同様の BMI デバイスを用いて、被験者の3種の意識(Rest, UP、DOWN)を読み取って識別する BMI システムを開発し、15 分間の訓練の後、全体的な識別率は最大75%であった。

また板井[14]らは、SSVEP を利用した BMI システムの構築のために、SSVEP の解析を行っている. 4 種類の刺激周波数で LED を点滅させ、点滅する LED を見た時の 3 秒間の脳波の解析データから得られた各周波数に対する SSVEP の平均検知率は 53~84%であった. 更に、与えた視覚刺激の周波数とその高調波の振幅スペクトルを足し合わせることで、検知率は 70~80%になった.



Figure 3-2 Emotiv EPOC

#### 3-4. 定常状態視覚誘発電位(SSVEP)

人が視覚的な外部刺激を見たときに、Fig.3-3に示した後頭部にある視覚野で生じる誘発電位のことを視覚誘発電位(Visual Evoked Potential: VEP)と呼ぶ。そして視覚刺激の提示頻度を上げる場合に生じる視覚誘発電位のことを特に定常状態視覚誘発電位(Steady State Visual Evoked Potential: SSVEP)と呼ぶ。提示頻度がある周波数の視覚刺激を被験者が受けているとき、被験者の EEG からは VEP の反応が提示頻度の周波数と同期し出現する。その時の EEG を周波数解析した結果から刺激に対する周波数成分とその高調波の振幅スペクトルが、他の周波数よりも強く出ることが知られている。このような結果が表れる脳波が SSVEPである。また、SSVEP はブレイン・マシン・インタフェース (BMI)で用いられるその他の事象関連電位に比べて、比較的安定していることが知られている。

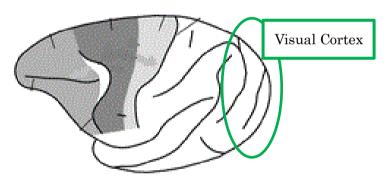


Figure 3-3 Place of the visual cortex

#### 3-5. 脳波の測定位置

本研究では、測定する頭部電極の基準となるグランド電極及び、リファレンス電極を両耳朶に設置した。測定に使用する電極は、視覚刺激を受け取る視覚野が後頭部に広く分布していることから後頭部に集中して装着した。測定位置は後頭部結節に電極 1 番を取り付け、頭頂部に向かって 20mm ほど上部に電極 2 番を取り付け、電極 2 番を中心に等間隔となるように他の三方向に電極 3 番から 5 番を配置した。この電極配置は拡張国際 10-20 法に基づき、測定点の最下点を後頭部結節と定め、後頭部を最小限の電極数で覆うためである。測定位置を Fig.3-4 に示す。

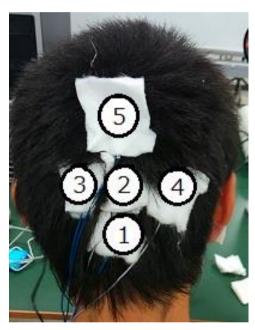


Figure 3-4 Measurement position of EEG

#### 第4章 予備実験

本章では本研究で用いた解析手法の説明を行い、視覚刺激に用いる動画の設定を決める 予備実験や、ある周波数の視覚刺激の動画によって起きる SSVEP の振幅スペクトルが与え た刺激周波数とその高調波の振幅スペクトルとして表れているか確かめる予備実験、 Fig.3-4 で示した電極数からより減らすことができないか確かめる予備実験について述べる.

#### 4-1. 本研究で用いた解析手法

本研究では、短時間フーリエ変換と加算平均法を用いている.

短時間フーリエ変換は、理論上無限の区間に渡って積分を行わなければならないフーリエ変換を行うために、範囲を区切った上でフーリエ変換する方法である。この範囲内は周期的で無限に繰り返されていると仮定して計算するのが一般的であるが、区切る際に範囲の端と端で不連続な要素ができるため、その区間に中央が1付近の値、その範囲外で0に収束する窓関数を掛けて不連続な要素を極力排除することが行われる。これが短時間フーリエ変換である。

$$F_{STFT}(t,\omega) = \int_{-\infty}^{\infty} f(t)\omega(\tau - t) \, e^{-i\omega t} \, dt \qquad (1)$$

※ωは窓関数の式

加算平均法とは、誘発電位など表れにくい反応を明瞭にする方法である. N 個の解析 データを加算し、平均することで特徴量を N 倍に、特徴量以外の値を $^1/_{\sqrt{N}}$  倍にする.

$$\mathbf{m} = \frac{\sum_{i=1}^{N} x_i}{N} \tag{2}$$

※Nはデータ個数

#### 4-2. 動画の色の組合せを決める実験

被験者が視覚刺激を受け取ったときに、どの点滅する図形の色と背景色の組合せがより SSVEP を強く出現させるかを調べるための実験を行った。実験では視覚刺激として作成した視覚刺激提示ソフトウェアを使用する。点滅する図形には四角を用い、図形の色は白と赤の2種類を用意し、背景色は黒と青を用意した。Fig.4-1 に示した図形の色と背景色の全ての組合せ(白黒、白青、赤黒、赤青)に対して、図形の点滅周波数を10Hz、15Hz(ソフト

ウェア上での設定, 10Hz:周期 100[ms], 15Hz:周期 66[ms]) の2種類を用意した.

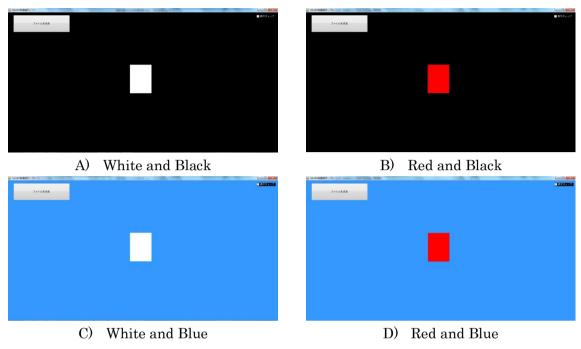


Figure 4-1 Color of blinking figure and background

実験では視覚刺激を 30 秒提示し、各 3 回分の EEG を測定した。実験時のサンプリング 周波数は、1kHz で測定した。なお、本実験での脳波測定における電極配置は、Fig.3-4 の電極 3 番、4 番の下側にそれぞれの電極間隔が 20mm となるように、さらに 2 個電極を追加し、電極 5 番は装着せず、計 6 個の電極で測定した。

脳波の解析手法は次のようにした. Fig.4-2 に示すように, EEG データの全区間 16384点 (約 16 秒間) に対して, データの解析区間を 2048点 (約 2 秒間) とした. データのオーバーラップは行わずに, 分割された 8 つの区間の EEG データそれぞれを短時間フーリエ変換 (STFT) し, 加算平均を行う.

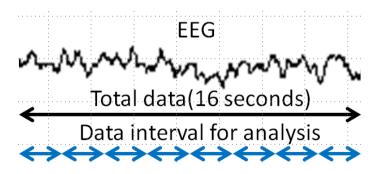


Figure 4-2 Method of data analysis for EEG

それぞれの条件 (色の組合せ 4 種, 点滅周波数 2 種, 電極 7 個) において測定された EEG に対して、前述のデータ解析を行った.各電極における解析結果から、Fig.4-3 のような与えた視覚刺激の周波数とその高調波に対する周波数付近にピークが現れ、明らかに SSVEP が現れたと考えられる電極の数をカウントし、どの程度 SSVEP が出現したかをまとめたものを Table 4-1 に示す.表内の数値は百分率表記である.

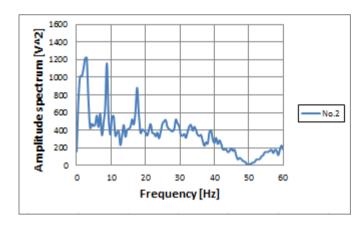


Figure 4-3 Analysis result of stimulus frequency 10Hz

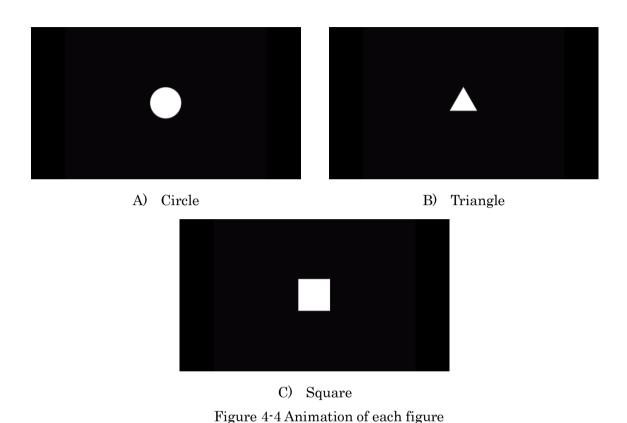
Color of blinking figure and background	White	White	Red	Red
	and	and	and	and
	Black	Blue	Black	Blue
10Hz	72.2	50.0	61.1	38.9
15Hz	27.8	55.6	11.1	22.2

Table 4-1 Appearance ratio of SSVEP in combination of color [%]

この結果から、刺激周波数 10Hz における白黒の組合せの時の出現率が 72.2%と最も高く、また白黒の組合せが他の組合せより図形と背景の境界を認識しやすいと考え、本研究では図形の色は白、背景色は黒に設定することにした.

#### 4-3. 図形の形を決める実験

被験者が視覚刺激を受け取ったときに、どのような図形が SSVEP を強く出現させるかを調べるための実験を行った。実験では視覚刺激として、30 秒間図形が点滅表示する動画を作成した。点滅させる図形には、Fig.4-4 のような丸、三角、四角の3 種類を用意し、点滅周波数は、8Hz、10Hz、12Hz、15Hz の4 種類とした。



この実験での電極は Fig.3-4 のように配置した. 実験では視覚刺激を 30 秒提示し、各 3 回分の EEG を測定した

実験では視覚刺激を 30 秒提示し、各 3 回分の EEG を測定した。実験時のサンプリング 周波数は、1kHz で測定した。

それぞれの条件(点滅図形 3 種,点滅周波数 4 種,電極 5 個)において測定された EEG に対して,4-1 節と同じデータ解析を行った.各電極における解析結果から,与えた視覚刺激の周波数とその高調波に対する周波数付近にピーク値が現れ,明らかに SSVEP が現れたと考えられる電極の数をカウントし,どの程度 SSVEP が出現したかをまとめたものを Table 4-2 に示す.表内の数値は百分率表記である.

Table 1 211ppearance race of SE , ET for each Similary ingere [70]						
	8Hz	10Hz	12Hz	$15 \mathrm{Hz}$	Average	
Circle	40.0	40.0	46.7	46.7	43.3	
Triangle	60.0	40.0	6.67	46.7	38.3	
Square	66.7	66.7	46.7	60.0	60.0	

Table 4-2 Appearance ratio of SSVEP for each blinking figure [%]

この結果から、本研究では点滅させる図形を一番出現率が高かった四角に設定することにした.

#### 4-4. SSVEP を検出するための周波数を調べる実験

被験者に視覚刺激を与えた時の EEG を周波数解析した際に、被験者の生理的な要因等により、必ずしも与えた視覚刺激の点滅周波数とちょうど同じ周波数に振幅スペクトルのピークが出るとは限らない。そこで、視覚刺激として用意した各点滅周波数に対して、SSVEPが現れたと考えられる周波数を調べる実験を行った。4-1、4-2 節の予備実験の結果に基づき、視覚刺激として、 Fig.2-3 に示すように黒の背景に白い四角の図形を 30 秒間点滅表示する動画を作成した。視覚刺激の点滅周波数は、8Hz、10Hz、12Hz、15Hz の 4 種類を用意し、実験時では動画を全画面表示とした。

実験では、以下に示す2つの条件で脳波の測定を行った.

- a) 白い四角の図形を表示したまま点滅させない状態
- b) 白い四角の図形を 8Hz, 10Hz, 12Hz, 15Hz で点滅させた状態

実験でのサンプリング周波数は 1kHz とし、椅子に座った状態で目から動画の画面までの距離を  $45\sim50cm$  とした体勢で実験を行った.

a)の点滅なしの状態では、被験者に図形を表示した状態の画面を注視してもらい、脳波が落ち着いた時点を測定開始とし、脳波を30秒間測定した.

b)の視覚刺激を提示する状態では、まず脳波の測定開始前に、被験者に測定対象とする周波数の点滅表示動画を3回見せて、視覚刺激に慣れてもらった。その後、脳波の測定を開始し、脳波が落ち着いた時点で30秒間の点滅表示動画を再生した。その際、視覚刺激を提示した時間が分かるように、動画の再生開始時と終了時の時刻を記録した。この作業を8Hz、10Hz、12Hz、15Hzの順で行い、点滅なしの状態と4種類の周波数による点滅表示を1セットとし、12セット分の脳波を測定した。

解析手法は次のように行った.解析の開始点を、点滅なしの状態では測定開始時刻、点滅させた状態では刺激提示開始時刻とする. EEG データの全区間は8192点(約8秒間), データの解析区間は2048点(約2秒間)とし、データのオーバーラップは行わない. 分割された4つの区間のEEG データそれぞれを短時間フーリエ変換(STFT)し、4個の解析データを用いて加算平均を行う. 各電極における解析結果から、与えた点滅周波数に対して、実際に振幅スペクトルのピークが現れた周波数とその高調波に対する周波数を読み取った. 12セット分の測定データに対して、ピークが現れた周波数帯とその高調波に対する周波数帯を電極毎にまとめたものをTable 4-3に示す.

	No.1	No.2	No.3	No.4	No.5
8Hz	$7.32 \sim 9.76$	$7.32 \sim 9.76$	$7.32 \sim 9.76$	6.83~8.78	7.32~9.27
Harmonics	15.1~16.6	16.1	15.6~16.1	15.6~16.1	15.1~16.6
10Hz	$9.76 \sim 10.7$	9.76	9.27~10.2	8.78~10.2	9.27~10.2
Harmonics	19.0~20.5	20.0	20.0	20.0	19.0~20.9
$12 \mathrm{Hz}$	11.7~12.2	11.7~12.2	11.7~12.2	11.7~12.6	11.7~12.2
Harmonics	22.9~23.9	23.9	23.9	23.9	23.4~24.9
$15 \mathrm{Hz}$	14.1~15.1	14.6~15.1	14.6~15.1	14.6~15.1	15.1
Harmonics	29.7~30.2	29.7~30.2	29.7~30.2	29.7~30.2	29.7~30.2

Table 4-3 Frequency band for appearance of SSVEP [Hz]

Table 4-3 に示すように、特に低周波数の点滅刺激(8Hz,10Hz)においては帯域幅が大きく、SSVEPが出現したと考えられる周波数が安定していないが、高周波数の点滅刺激(12Hz,15Hz)においては、刺激を与えた周波数とほぼ同じ周波数において SSVEP が出現していると考えられる。そこで本研究では、各点滅周波数に対して、SSVEP を検出するために振幅スペクトルを求める周波数を、与えた点滅周波数(8Hz,10Hz,12Hz,15Hz)とその高調波(16Hz,20Hz,24Hz,30Hz)と同じ値とすることにした。

#### 4-5. 電極を減らすための実験

被験者に対する負担を考えると電極数は少ない方が好ましいので、Fig.3-4の電極数よりも電極数を減らすための実験を行った。4-3節で測定した EEG データを用いて、4-3節と同じ解析手法により、分割された 4つの区間の EEG データそれぞれを STFT し、被験者に与えた 8Hz、10Hz、12Hz、15Hz の刺激周波数とその高調波の振幅スペクトルの値を足し合わせた。

そして、足し合わせたそれぞれの振幅スペクトルを解析開始時間から解析終了時間まで 時系列順に並べてグラフ化し、点滅なしの状態とそれぞれの刺激周波数で点滅させた状態 のグラフを比較する.

それぞれの条件(点滅なし及び点滅周波数 4 種,電極 5 個)において測定された EEG に対して,上記の解析を行った.一例として,電極 2 番で測定した 9 セット目の測定データにおける点滅なしの状態の 8Hz とその高調波の振幅スペクトルを足し合わせた数値と,点滅周波数 8Hz の視覚刺激を与えた時の 8Hz,10Hz,12Hz,15Hz とそれぞれの高調波の振幅スペクトルを足し合わせた数値を Fig.4-5 に示す.

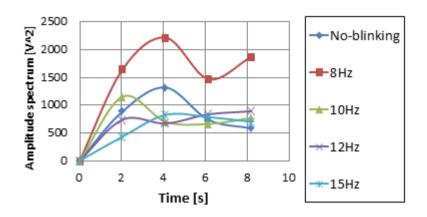


Figure 4-5 Amplitude spectrum for each frequency

他の電極や他のセットの測定データにおいても、Fig.4-5に示されるように、与えた刺激 周波数とその高調波の振幅スペクトルの和が、点滅なしの状態よりも大きく、かつ与えた 刺激周波数ではない周波数とその高調波の振幅スペクトルの和よりも大きくなることが概 ね観測できた.これにより、電極数を減らすために上記の現象がより顕著に現れる電極を 調べた.

各電極における1セットから12セットまでの解析結果において、与えた刺激周波数とその高調波の振幅スペクトルの和が他の周波数よりも連続して2点以上大きくなっている解析結果の数を電極毎にカウントする. ただし、下記の条件に当てはまる場合はカウントしないこととする.

- i) 与えた刺激周波数ではない周波数が先にカウントの条件を満たし、その後に与えた刺激周波数がカウントの条件を満たした場合
- ii) 与えた刺激周波数における振幅スペクトルの和と他の周波数における振幅スペクトルの和の大小が目視で確認できない場合

感度の指数は、上記の条件から得られた各電極におけるカウント数を全 12 回で除することにより求めた、電極毎の刺激周波数に対する感度を Table 4-4 に示す. Table 内の数値は百分率表記である.

Table 4-4 Sensitiveness of each electrode for stimulus frequency [%]

	No.1	No.2	No.3	No.4	No.5
8Hz	8.33	33.3	66.7	83.3	33.3
10Hz	50.0	75.0	58.3	75.0	41.7
12Hz	66.7	91.7	83.3	100	25.0
15Hz	25.0	75.0	91.7	100	8.33
Total	37.5	68.8	75.0	89.6	27.1

この結果から、電極 1 番と 5 番は他の電極に比べて感度が低いため除外し、電極 2 番、4 番を 5 章の SSVEP 検出のための電極として採用することにした.

## 第5章 SSVEPの検出実験

4章の予備実験の結果から、視覚刺激に用いる動画の設定は、黒の背景に白い四角の図形とし、SSVEPの検出実験での電極配置は、Fig.3-4の電極2番、3番、4番の3個に選定した。これらの実験環境を基にSSVEPの検出実験を行った。

#### 5-1. SSVEP の検出条件

短時間で SSVEP を検出するための脳波の解析手法を以下のように定める. ただし, 4-4 節の実験結果から, 点滅なしの状態は解析に影響を与えないと考えたため, 本実験においては考慮しないものとする.

解析の条件を以下のように定める. 解析の開始点は、刺激提示開始時刻とする. EEG データの全区間を 6144 点(約 6 秒間)、データの解析区間を 2048 点(約 2 秒間)とする. データのオーバーラップは 1792 点(約 1.75 秒間)とする. Fig.5-1 に示す 8 つの区間の EEG データそれぞれを STFT し、それぞれにおいて 8Hz、10Hz、12Hz、15Hz の刺激周波数とその高調波の振幅スペクトルを足し合わせた. そして各周波数に対して、8 つの区間 毎に移動平均を求めた. それぞれの条件(点滅周波数 4 種、電極 3 個)に対して、4-3 節の実験で測定した EEG データを用いて、上述した解析を行う.

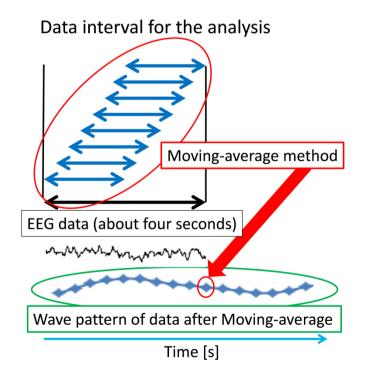
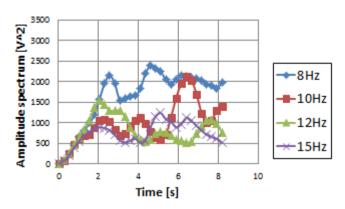
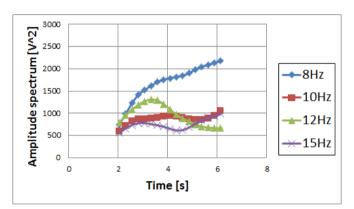


Figure 5-1 Method of data analysis for detecting SSVEP in short time

足し合わせたそれぞれの周波数の振幅スペクトルを解析開始時間から解析終了時間まで時系列順に並べてグラフ化した. 一例として, 点滅周波数 8Hz の視覚刺激を与えた時の電極 3 番で測定した 3 セット目の測定データにおける脳波での移動平均適用前と適用後の解析結果を Fig.5-2 に示す.



a) Before the application of the moving average method



b) After the application of the moving average method Figure 5-2 Result of data analysis

Fig.5-2-bより、移動平均法を適用後の結果を見ると、与えた視覚刺激の周波数 8Hz とその高調波の振幅スペクトルの和が他の周波数の振幅スペクトルの和よりも大きく現れていることがわかる。また移動平均を行うことにより、上述の解析手法において移動平均を行わずに同じ脳波の測定データを解析した結果よりも、この現象が明確に現れることを確認している。

以上の結果に基づいて、SSVEPの検出条件を以下のように定める. 前述の解析手法で得られた移動平均を適用した各電極における解析結果 (例えば Fig.5-2-b) において、与えた刺激周波数とその高調波の振幅スペクトルの和が他の周波数よりも連続して 4 点以上大きくなる電極のデータを SSVEP が出現したと考え、SSVEP が出現した電極の数をカウントする. カウントしない条件は 4-4 節で記した i)、ii)の条件と同じとする. そして、3 個の

電極のうち、2個以上の電極がカウントされれば、その測定データにおいて与えた点滅周波 数に対する SSVEP が検出できたと定義する.

#### SSVEP の検出実験結果 5-2.

5-1 節で定めた SSVEP の検出条件を 4-3 節で測定した EEG データに適用して、提案し た検出方法により SSVEP を検出できるかを検証した. それぞれの刺激周波数に対して SSVEP が解析全区間 6 秒以内に検出できたかどうか、及びその時に SSVEP が出現したと してカウントされた電極数を Table 5-1 に示す. 検出できた場合は○, 検出できなかった場 合は×で表し、括弧内の数値が SSVEP 出現としてカウントした電極数を表している.

また、SSVEP を検出できた時において刺激提示から  $2\sim4$  秒、 $2\sim6$  秒の時間ごとの SSVEP の累積検出率を Table 5-2 に示す. 表内の数値は百分率表記である.

Table 5-1 Experimental result of detecting SSVEP [%]

	8Hz	10Hz	12Hz	15Hz
$1^{\mathrm{st}}$	0 (3)	× (0)	0 (3)	0 (3)
$2^{ m nd}$	0 (2)	0 (3)	0 (3)	0 (3)
$3^{\rm rd}$	0 (3)	0 (2)	0 (3)	0 (3)
$4^{ m th}$	0 (2)	0 (3)	0 (3)	0 (3)
$5^{ m th}$	× (1)	0 (3)	0 (3)	0 (3)
$6^{ m th}$	0 (3)	0 (3)	0 (3)	0 (3)
$7^{ m th}$	0 (3)	0 (2)	0 (3)	0 (3)
8 <sup>th</sup>	0 (2)	× (1)	0 (3)	0 (3)
$9^{\mathrm{th}}$	0 (3)	0 (3)	0 (3)	0 (3)
$10^{ m th}$	0 (3)	× (0)	0 (3)	0 (3)
$11^{ m th}$	× (0)	0 (3)	0 (3)	0 (3)
$12^{ m th}$	× (0)	0 (3)	0 (3)	0 (3)

Table 5-2 Accumulated detection rate of SSVEP [%]

	8Hz	$10 \mathrm{Hz}$	$12 \mathrm{Hz}$	$15 \mathrm{Hz}$
$2[s]\sim 4[s]$	33.3	33.3	91.7	75.0
$2[s] \sim 6[s]$	75.0	75.0	100	100

#### 5-3. 考察

Table 5-1 の SSVEP の検出結果より、SSVEP 検出条件を用いた場合では実験回数 12 回中 8Hz, 10Hz は 9回,12Hz, 15Hz は 12回検出できた. 特に 12Hz, 15Hz は全実験で 3個の電極全てがカウントされた. また、Table 5-2 より、8Hz、10Hz では 2~6 秒間で検出率 75%、12Hz、15Hz では 2~6 秒間で検出率 100%となっており、高周波数ほど速く高い検出率が得られている. 低周波数においては、本研究の被験者は特に 8Hz の点滅周波数に対して高い検出率が得られなかった.

低周波数において高い検出率が得られていない理由として、Table 4-3 に示したように、低周波数 (8Hz, 10Hz) において SSVEP が出現したと考えられる周波数が安定していないことが挙げられる。したがって、SSVEP が安定して出現できるように 8Hz, 10Hz の動画をより細かく設定を決定することや各点滅周波数に対して、SSVEP を検出するために振幅スペクトルを求める周波数を、与えた点滅周波数 (8Hz,10Hz,12Hz,15Hz) とその高調波(16Hz,20Hz,24Hz,30Hz)と同じ値とするだけではなく、8Hz, 10Hz の場合ではその周波数付近の振幅スペクトルを考慮に入れることで、低周波数に対する検出率を向上できる可能性がある。

#### 第6章 複数の光刺激提示下での SSVEP の識別

実際に SSVEP によって複数のコマンドの選択が可能であるか調べるために、刺激周波数の異なる 2 種及び 3 種の視覚刺激を与えながら、被験者が一方を注視した時、注視した視覚刺激の影響を受けてその視覚刺激の周波数の SSVEP が識別できるかどうか検証した.

#### 6-1. 2 種の点滅動画による SSVEP の識別実験

刺激周波数の異なる 2 種の視覚刺激の動画を用意し、どちらか一方の動画を注目することによって注目した視覚刺激の周波数の SSVEP が識別できるかどうか実験を行った.

実験における動画の設定と選定した電極配置やサンプリング周波数,使用する点滅周波数の種類は5-1節と同じ条件とした.また画面照度を相対的に強めるために,モニターと被験者の上部に覆いを設けて, Fig.6-1 のような画面付近が薄暗がりになるような環境を用意した.

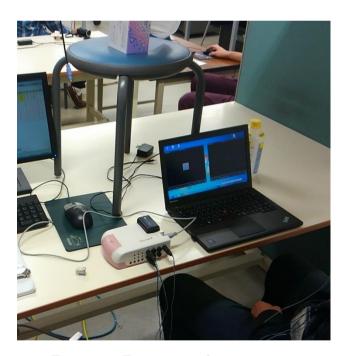


Figure 6-1 Experimental environment

被験者は椅子に座った状態で顔から画面までの距離が  $45\sim50$ cm となるような体勢で実験を行った。また、画面は Fig.6-2 のような配置で動画をループ再生し、注目させる動画は左側に配置した。

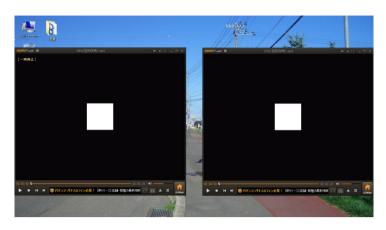


Figure 6-2 Appearance of the screen under two stimulus presentations

実験では画面を完全に隠した状態にして、脳波が落ち着いた時点を測定開始とし、測定者が任意のタイミングで画面の遮蔽物を取り除いた。画面が完全に露出した時刻を刺激提示開始時刻とし、約15秒間刺激を与え、刺激提示後の時刻を刺激提示終了時刻とし、測定終了とした。注目する刺激周波数の動画とその他の周波数の動画の組合せ3種類を1セットとし、7セット分測定した後、注目する刺激周波数の動画を変更し、同様に測定した。なお、この実験は2日に分けて行った。

#### 6-2. 2種の SSVEP の識別条件

実験の解析における STFT の解析条件や SSVEP の検出条件については 5-1 節と同じとし、 EEG データの解析全区間は 10240 点(約 10 秒間)とした。また解析結果を元に Fig.5-2-b のようなグラフを各セット,電極毎に作成し,測定日毎に最も良いデータと最も悪いデータを測定日毎で取り除き,母数 10 個のデータとして識別率を求めた。また 5 章までは目視による判断であったため, SSVEP の識別を自動化するために,数値的判断を MATLAB/Simulink を用いた SSVEP 識別プログラムによって行った。なお,ある周波数の振幅スペクトルと他の周波数の振幅スペクトルの大小を判定する際に,振幅スペクトルの差が 100 以上となった時(Fig.16 では 6 秒前後での 12Hz と 15Hz)に明確に大小関係が現れたと判断し,SSVEP 検出のためのカウントに含めることにした。

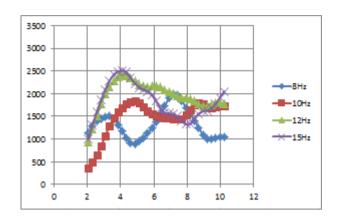


Figure 6-3 Case of the judgment by the MATLAB/Simulink program

### 6-3. 2種の点滅動画による SSVEP の識別実験結果

注目する刺激周波数毎の識別結果を Table 6-1 に示す。識別できた場合は $\bigcirc$ 、識別できなかった場合は $\times$ で表し、括弧内の周波数が誤識別時の周波数を表している。表中の識別率は  $2\sim10$  秒に対してのものである。

Table 6-1 Distinction results under two stimulus presentations

#### a) Stimulus frequency 8Hz gaze

	, 011111 011 012 11 0 0	1 ,	<b>9</b>
	8Hz-10Hz	8Hz-12Hz	8Hz-15Hz
Day1,1st	0	× (10Hz)	0
Day1,2 <sup>nd</sup>	× (12Hz)	0	0
Day1,3rd	0	0	0
Day1,4th	× (12Hz)	× (12Hz)	× (10Hz)
Day1,5th	× (12Hz)	0	0
Day2,1st	× (12Hz)	× (10Hz)	× (12Hz)
Day2,2 <sup>nd</sup>	0	0	0
Day2,3rd	× (10Hz)	× (10Hz)	0
Day2,4th	0	0	× (10Hz)
Day2,5th	0	0	0
Distinction	50.0%	60.0%	70.0%
rate	90.0%	00.0%	70.0%

#### b) Stimulus frequency 10Hz gaze

	10Hz-8Hz	10Hz-12Hz	10Hz-15Hz
Day1,1st	0	0	0
Day1,2nd	0	0	0
Day1,3rd	× (12Hz)	0	0
Day1,4th	0	0	× (12Hz)
Day1,5th	0	0	0
Day2,1st	0	0	0
Day2,2nd	× (15Hz)	× (12Hz)	0
Day2,3rd	0	0	0
Day2,4th	0	0	0
Day2,5th	× (12Hz)	0	0
Distinction rate	70.0%	90.0%	90.0%

#### c) Stimulus frequency 12Hz gaze

	12Hz-8Hz	12Hz-10Hz	12Hz-15Hz	
Day1,1st	× (8Hz)	0	0	
Day1,2nd	0 0		0	
Day1,3rd	0	0	0	
Day1,4th	0	0	0	
Day1,5 <sup>th</sup>	0	0	0	
Day2,1st	0	0	× (10Hz)	
Day2,2nd	0	0	0	
Day2,3rd	0	0	0	
Day2,4th	0	0	0	
Day2,5th	0	0	0	
Distinction	90.0%	1000/	00.00/	
rate	90.0%	100%	90.0%	

d) Stimulus frequency 15Hz gaze

	15Hz-8Hz	15Hz-10Hz	15Hz-12Hz
Day1,1st	0	0	× (12Hz)
Day1,2nd	0	0	0
Day1,3rd	× (12Hz)	× (10Hz)	× (12Hz)
Day1,4th	0	× (8Hz)	0
Day1,5 <sup>th</sup>	0	0	0
$\mathrm{Day2}, 1^{\mathrm{st}}$	× (12Hz)	0	0
Day2,2nd	0	0	0
Day2,3rd	× (12Hz)	0	0
Day2,4th	× (12Hz)	0	0
Day2,5th	× (12Hz)	0	0
Distinction rate	50.0%	80.0%	80.0%

また、注目した動画と組み合わせた動画での識別率が 80.0%以上であった場合を $\bigcirc$ で示したものを Table 6-2 を示した。表 8 において、縦方向は注目する刺激周波数、横方向は組合せ相手の刺激周波数である.

Table 6-2 Correlation of distinction result for all combinations

	8Hz	10Hz	12Hz	15Hz
8Hz		×	×	×
10Hz	×		0	0
12Hz	0	0		0
15Hz	×	0	0	

Table 6-1 より、8Hz を除いた 10Hz、12Hz、15Hz の組合せでは識別率は 80%以上であった。また Table 6-2 に示す通り、8Hz が組み合わさると識別率が他の組合せよりも低下するため、次節の 3 種の動画による SSVEP の識別実験では、10Hz、12Hz、15Hz の動画を用いることにした。

### 6-4. 3種の点滅動画による SSVEP の識別実験

刺激周波数の異なる3種の視覚刺激の動画を用意し、一つの動画を注目することによって SSVEP が識別できるかどうか実験を行った.実験環境や、測定条件などは6-1節と同じとした.また、画面は Fig.6-4 のような配置で動画をループ再生し、注目させる動画は上部に配置し、組合せ相手となる視覚刺激の動画は左側が小さい周波数、右側が大きい周波数となるようにした.

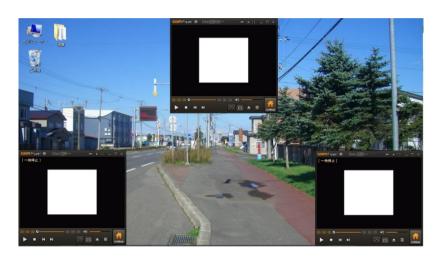


Figure 6-4 Appearance of the screen under three stimulus presentations

実験では画面を完全に隠した状態にして、脳波が落ち着いた時点を測定開始とし、測定者が任意のタイミングで画面の遮蔽物を取り除いた。画面が完全に露出した時刻を刺激提示開始時刻とし、約15秒間刺激を与え、刺激提示後の時刻を刺激提示終了時刻とし、測定終了とした。注目する刺激周波数の動画とその他の周波数の動画の組合せ3種類を1セットとし、7セット分測定した後、注目する刺激周波数の動画を変更し、同様に測定した。なお、この実験も二日に分けて行い、SSVEPの検出条件は、6-2節と同じとした。

### 6-5. 3種の点滅動画による SSVEP の識別実験結果

注目する刺激周波数毎の識別結果を Table 6-3 に示す. 識別できた場合は $\bigcirc$ , 識別できなかった場合は $\times$ で表し、括弧内の周波数が誤識別された周波数を表している. また、SSVEPを識別できた時において、刺激提示から  $2\sim4$  秒、 $2\sim6$  秒、 $2\sim8$  秒、 $2\sim10$  秒の時間ごとの SSVEP の累積識別率を表 6-4 に示す. 表内の数値は百分率表記である.

Table 6-3 Distinction results under three stimulus presentations

	10Hz gaze	12Hz gaze	15Hz gaze
Day1,1st	× (15Hz)	0	0
Day1,2 <sup>nd</sup>	0	0	0
Day1,3rd	0	0	× (12Hz)
Day1,4 <sup>th</sup>	0	0	0
Day1,5 <sup>th</sup>	0	0	0
Day2,1st	0	0	0
Day2,2 <sup>nd</sup>	0	0	× (12Hz)
Day2,3rd	0	0	× (10Hz)
Day2,4th	0	× (15Hz)	0
Day2,5 <sup>th</sup>	× (12Hz)	0	0
Distinction rate	80.0%	90.0%	70.0%

Table 6-4 Accumulated distinction rate of SSVEP [%]

	10Hz gaze	12Hz gaze	15Hz gaze
$2[s] \sim 4[s]$	80.0	80.0	30.0
$2[s] \sim 6[s]$		80.0	50.0
$2[s] \sim 8[s]$		90.0	60.0
$2[s] \sim 10[s]$			70.0

Table 6-3 から 10Hz, 12Hz の識別率は 80.0%以上であり、Table 6-4 から 2~6 秒間では 80.0%の識別率であった。15Hz の識別率は 2~10 秒間では 70.0%であるが、 2~6 秒間では 50.0%と低かった。各周波数の識別率を高くし、かつ均等にするために、15Hz の動画の設定として、図形の大きさを他の動画より一回り大きく、また Table 4-1 から 15Hz の出現率が高い色の組合せとなるように、図形の色を白、背景色を青に変更し、再度同様の実験を行った。同様に解析を行い、この変更の下での注目する刺激周波数毎の識別結果をTable 6-5 に示す。識別できた場合は $\bigcirc$ 、識別できなかった場合は $\bigcirc$ 、表し、括弧内の数字は識別、誤識別が行われた時間帯を表している。

Table 6-5 Distinction results after change of animation for 15Hz

	10Hz gaze	12Hz gaze	15Hz gaze
Day1,1st	0 (2~4)	0 (2~4)	o <b>(2~4)</b>
Day1,2 <sup>nd</sup>	0 (2~4)	0 (2~4)	o <b>(2~4)</b>
Day1,3rd	0 (2~4)	0 (2~4)	o <b>(2~4)</b>
Day1,4th	0 (2~4)	0 (2~4)	o <b>(2~4)</b>
Day1,5 <sup>th</sup>	0 (2~4)	0 (2~4)	0 (2~4)
Day2,1st	0 (2~4)	0 (2~4)	0 (2~4)
Day2,2 <sup>nd</sup>	0 (2~4)	0 (2~4)	0 (2~4)
Day2,3rd	0 (2~4)	0 (2~4)	o <b>(2~4)</b>
Day2,4th	0 (2~4)	0 (2~4)	o <b>(2~4)</b>
Day2,5 <sup>th</sup>	0 (2~4)	0 (2~4)	0 (2~4)
Distinction rate	100%	100%	100%

全ての識別結果において、注目した刺激周波数が対して正しく識別された。また、SSVEPと識別された時間は、全て  $2\sim4$  秒間で行うことができた。

これらの実験結果より、単独で 15Hz の視覚刺激動画を提示した際には早く正確に SSVEP が検出できたが、他の視覚刺激動画と組み合わせて提示すると互いに干渉して被験 者に影響を与え、識別率が低下することが分かった。しかしながら、被験者の特性に沿って 15Hz の視覚刺激動画を修正すると、識別率および識別時間ともに改善できることが確認できた.

### 6-6. 考察

2種の点滅動画による SSVEP の識別実験結果をまとめた Table 6-1 より、8Hz を除いた 10Hz、12Hz、15Hz の組合せでは識別率は 80.0%以上であった。また Table 6-2 に示す通り、8Hz が組み合わさると識別率が他の組合せよりも低下する。これは 8Hz の動画のみ 24fps での設定で作成したため、60fps の動画と比べると 24fps の 1 フレームの時間長は約 0.042(1/24)[s]、60fps の 1 フレームの時間長は約 0.017(1/60)[s]と 2.5 倍の時間差がある。そのため 8Hz 点滅の動画は視覚刺激として被験者に影響を与えにくくなったと考えた。

また 2 種の点滅動画の識別実験結果を受けて、10Hz、12Hz、15Hz の 3 種の点滅動画による SSVEP の識別実験は、結果として Table 6-3 から 10Hz、12Hz の識別率は 80.0%以上であり、Table 6-4 から 2~6 秒間では 80.0%の識別率であった。15Hz の識別率は、2~10 秒間では 70.0%であるが、 2~6 秒間では 50.0%と低かった。この結果を受け、各周波数の識別率を高くし、かつ均等にするために、15Hz の動画の設定として、図形の大きさを他の動画より一回り大きく、また Table 4-1 から 15Hz の動画の設定として、図形の大きさまうに、図形の色を白、背景色を青に変更し、再度同様の実験を行った。その結果、全ての識別結果において、注目した刺激周波数に対して正しく識別された。また SSVEP と識別された時間は、全て 2~4 秒間で行うことができた。これらの実験結果より、単独で 15Hzの視覚刺激動画を提示した際には早く正確に SSVEP が検出できたが、他の視覚刺激動画と組み合わせて提示すると互いに干渉して被験者に影響を与え、識別率が低下することが分かった。しかしながら、被験者の特性に沿って 15Hz の視覚刺激動画を修正すると、識別率および識別時間ともに改善できることが確認できたため、各刺激周波数の視覚刺激に個別の設定をすることで、被験者により早く、より影響を与えることができると考えた。

## 第7章 3種の点滅動画によるランダム注視実験

第6章の実験結果より、複数の光刺激提示下でのSSVEPの識別は可能であると考えた. そこで 3種の視覚刺激提示用動画を決まった配置で固定し、被験者に測定者のランダムに指定する動画に注視してもらった. そして測定した脳波を解析し、注視した視覚刺激の影響を受けてその視覚刺激の周波数のSSVEPが識別できるかどうか検証した.

### 7-1. 3種の点滅動画によるランダム注視実験

実際に SSVEP によって複数のコマンドの選択が可能であるかを調べるために、刺激周波数の異なる 3 種の視覚刺激の動画を用意し、一つの動画を注目することによって SSVEP が識別できるかどうか実験を行った. 実験環境や、測定条件は 6-1 節と同様に用意した. また、動画の設定は 6-5 節の変更後の設定を適用し、画面は Fig.7-1 のような配置で動画をループ再生し、15Hz の動画を上部に、10Hz を左側、12Hz を右側に配置した.

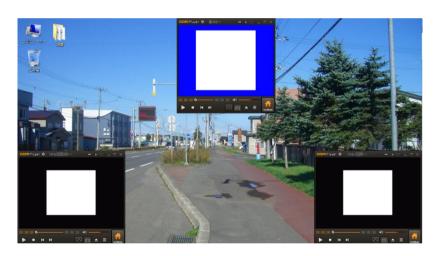


Figure 7-1 Appearance of final screen

実験ではあらかじめ、注目する刺激周波数 10Hz, 12Hz, 15Hz がランダムかつ 10 個ずつになる注目指示表を作成した. 測定開始前では画面を完全に隠した状態にして、脳波が落ち着いた時点を測定開始とし、測定者が注目指示表に従い任意のタイミングで被験者に注目させる周波数を告知し、その 3 秒後に画面の遮蔽物を取り除いた. これは被験者の体動などによるノイズが混入しないようにするためである. 画面が完全に露出した時刻を記録し、約 15 秒間刺激を与え、刺激提示後の時刻を記録し、測定終了とした. この手順を各周波数 10 個、計 30 個測定した. SSVEP の検出条件は、5.2 節と同じとした.

# 7-2. 3種の点滅動画によるランダム注視実験結果

注目する刺激周波数毎の識別結果を Table 7-1 に示す. 識別できた場合は $\bigcirc$ , 識別できなかった場合は $\times$ で表し、括弧内の周波数が誤識別された周波数を表している. また、SSVEPを識別できた時において、刺激提示から  $2\sim4$  秒、 $2\sim6$  秒ごとの SSVEP の累積識別率を Table 7-2 に示す. 表内の数値は百分率表記である.

Table 7-1 Distinction results under three stimulus presentations by random gaze

	10Hz gaze	12Hz gaze	15Hz gaze
1 <sup>st</sup>	0	0	0
$2^{ m nd}$	0	0	0
$3^{\mathrm{rd}}$	0	0	0
$4^{ m th}$	0	0	0
$5^{ m th}$	0	0	0
$6^{ m th}$	0	0	0
$7^{ m th}$	0	0	0
8 <sup>th</sup>	0	× (10Hz)	0
9 <sup>th</sup>	× (15Hz)	0	0
$10^{ m th}$	0	0	0
distinction rate	90.0%	90.0%	100%

Table 7-2 Accumulated distinction rate of SSVEP by the random gaze [%]

	10Hz gaze	12Hz gaze	15Hz gaze
$2[s] \sim 4[s]$	70.0	70.0	100
2[s]~6[s]	90.0	90.0	

### 7-3. 考察

Table 7-1の SSVEP の識別結果より、全ての刺激周波数に対して SSVEP を 90.0%以上 識別できた. また、Table 7-2 より、2~6 秒間で識別率 90.0%以上となっており、被験者の 脳波に SSVEP の現象が表れる時間が早くなった. この結果から被験者がランダムに注視しても識別率が高く、早く識別できているため、SSVEP によって 3 種のコマンドの選択が可能であると考えられる.

但し、常時点滅した画面状態において被験者が注視していない状態から被験者が注視した状態に遷移した時の変化を実験にて確認していないため、実際に BMI として適用するためには、この検出方法に新たな識別条件を加える必要がある可能性がある。また電動車椅子の BMI として使用するためには走行時や停止時に起こると思われる体動ノイズなどを除去する必要があるため、同様にノイズの除去条件を加える必要があると考えた。

## 第8章 結論

本研究では、BMI に使用できる脳波解析の基礎研究として、定常状態視覚誘発電位 (SSVEP) を短時間で検出する方法を提案し、その検出方法を用いて複数の点滅視覚刺激環境下における SSVEP の識別を行った.予備実験を行い、視覚刺激として与える点滅図形を白い四角、背景色を黒、使用する電極数を 3 個とした.また、SSVEP を検出するための脳波の解析手法、及び検出条件を定め、8Hz、10Hz、12Hz、15Hz で点滅する刺激動画を用いて、単一の視覚刺激による SSVEP の検出実験を行った.6 秒間での各周波数の検出率は 75~100%となり、特に高周波数に対する検出率が高かった.

また複数の光刺激提示下でのSSVEPの識別実験を行った. 2種の点滅動画によるSSVEPの識別実験結果は、8Hzを除いた 10Hz、12Hz、15Hz、の注視結果は 80%以上の識別率であった。また 8Hz が組み合わさると識別率が他の組合せよりも低下するため、3種の動画による SSVEP 識別実験では、10Hz、12Hz の動画を用いた。3種の点滅動画による SSVEPの識別実験を行った結果、 $2\sim6$  秒間では 10Hz、12Hz の識別率は 80.0%であったが、15Hz の識別率は 50.0%であった。よって各周波数の識別率を高くし、かつ均等にするために、15Hz の動画を改めて設定した。15Hz の動画の図形を一回り大きく、また動画の色の組合せを決める予備実験から 15Hz の出現率が高い色の組合せとなるように図形の色を白、背景色を青に設定した。この変更の下での 3種(10Hz、12Hz、15Hz)の点滅動画による SSVEPの識別実験結果は、 $2\sim4$  秒間において 100%の識別率となり、改善された。

最後に、各動画の配置を固定した上でランダムに注視しても、全ての刺激周波数に対して SSVEP を 90.0%以上識別できた. また、2~6 秒間で識別率 90.0%以上となっており、被験者の脳波に SSVEP の現象が表れる時間が早くなった. この結果から被験者がランダムに注視しても識別率が高く、早く識別できているため、SSVEP によって 3 種のコマンドの選択が可能であると考えられる.

但し、3種の点滅動画によるランダム注視実験時の実験条件下において常時点滅した画面状態において被験者が注視していない状態から被験者が注視した状態に遷移した時の変化を実験にて確認していないため、実際に BMI として適用するためには、この検出方法に新たな識別条件を加える必要がある可能性がある。また電動車椅子の BMI として使用するためには走行時や停止時に起こると思われる体動ノイズなどを除去する必要があるため、同様にノイズの除去条件を加える必要があると考えた。また、提案した SSVEP の識別方法を用いれば、被験者の特性に対応した視覚刺激の設定を行った上で、様々な機器を操作する BMI として利用できる可能性がある。

今後の課題として以下が挙げられる.被験者が注視していない時から注視した時の遷移の変化を提案した SSVEP 検出方法によって確認できるかの実験と、本実験では被験者が 1人であったため、今後は被験者数を増やして提案した SSVEP の検出方法の有用性を確認し、識別が可能であるか確かめる必要がある. さらに、実際に BMI として電動車椅子の操縦ができるかを確認する必要がある.

## 謝辞

本研究に関して終始熱心なご指導ご鞭撻を頂きました法政大学教授の石井千春教授に心より感謝致します。また本研究の実験を行うにあたり、被験者を快く引き受けて下さった小西氏に感謝致します。そして経過報告会では、多くの意見や示唆を頂きました医療・福祉ロボティクス研究室の同期、後輩の皆様に感謝の意を表します。

最後に大学生活を続けていく過程でご支援いただいた家族,友人の方へ心から感謝の気持ちと御礼を申し上げたく,謝辞にかえさせていただきます.

## 参考文献

- 1) http://www.nanbyou.or.jp/entry/52
- 2) http://www.nanbyou.or.jp/entry/4147
- 3) https://doctors-me.com/doctor/trauma/30
- 4) http://www.aist.go.jp/digbook/openlab/2012/#page=175
- 5) http://jp.diginfo.tv/v/12-0199-d-jp.php
- 6) http://spring-fragrance.mints.ne.jp/aviutl/
- 7) S.-P., Kim, J. D. Simeral, L. R. Hochberg, J. P. Donoghue, G. M. Friehs, and M. J. Black, "Multi-state decoding of point-and-click control signals from motor cortical activity in a human with tetraplegia,", Proc. of the 3rd International IEEE EMBS Conference on Neural Engineering, pp. 486~489, 2007.
- 8) L. R. Hochberg, at el., "Neuronal ensemble control of prosthetic devices by a human with tetraplegia", Nature Vol.442, pp.164-171, 2006.
- 9) M. Hajibabazadeh, V. Azimirad, "Brain-Robot interface: distinguishing left and right hand EEG signals through SVM", Proceeding of the 2<sup>nd</sup> RSI/ISM International Conference on Robotics and Mechatronics, October 15-17,2014, Tehran, Iran
- T. Noda, N. Sugimoto, J. Furukawa, M. Sato, S. Hyon, J. Morimoto, "Brain-Controlled Exoskeleton Robot for BMI Rehabilitation", Humanoid Robots (Humanoids), 12th IEEE-RAS International Conference,pp21-27, 2012
- M. Mano and G. Capi, "Adaptive navigation of a brain controlled robotic wheelchair in an indoor environment", Proc. of the 2013 JSME Conference on Robotics and Mechatronics, No.13-2, 2013.
- 12) 茨木仁希, 中村恭之, "脳信号収集ワイヤレスヘッドセットを用いた電動車椅子ロボットの制御", Proceedings of the 2013 JSME Conference on Robotics and Mechatronics, No.13-2, 2013.
- T. Jiralerspong, C.Liu, J.Ishikawa, "Identification of Three Mental States Using a Motor Imagery Based Brain Machine Interface", Computational Intelligence in Brain Computer Interfaces (CIBCI), 2014 IEEE Symposium
- 14) 板井陽俊,船瀬新王, "BCI システムの構築を目指した単一試行脳波の解析技術", 日本神経回路学会誌, vol.9, No.3, pp.118-125, 2012.

# 付録

# A) 使用機器使用

## A-1) 視覚刺激提示用ノートPC

#### ThinkPad X240

本研究では被験者に視覚刺激を与えるために Lenovo 社の ThinkPad X240 を用いた. 本研究で用いたノート PC の仕様を以下に示す.

Table A-1 Spec. of ThinkPad X240

	of Tillinki au A240		
製品番号		20AMA3H5JP	
開発製造元		レノボ・ジャパン株式会社	
OS		Windows 7 Enterprise Service Pack 1	
プロセッサ		Intel® Core™ i3-4010U CPU @1.70GHz	
システムの種類		64bit OS	
実装メモリ(RAM)		4.00GB (3.69GB 使用可能)	
ハードディスクドライブ	Cドライブ	空き領域 78.9GB/229GB	
	Dドライブ	空き領域 210GB/236GB	
ディスプレイ		LED バックライト付き	
		12.5 型 HD TFT 液晶	
		(1366×768 ドット, 1677 万色), 光沢無し	
本体寸法(W×D×H)mm		約 305.5x208.5x19.9-20.3mm	
本体質量 (バッテリー. パックを含む)		約 1.46kg	
バッテリー. パック 種類		3セル+3セル	
		リチウムイオン システム・バッテリー	
使用時	謂	約 11.6 時間	
充電時	計間	約3.4時間 (パワーオフ/サスペンド)	
		約 3.6 時間(使用時)	
最大消費電力(W)		45	
標準時消費電力(W)(アイドル時消費電力)		5.4	

## A-2) 脳波計測機器

### 脳波計 PolymateII

本研究では EEG (脳波) の計測にデジテックス研究所の PolymateII を用いた. 本研究で用いた脳波計の仕様を以下に示す.

Table A-2 Spec. of PolymateII

AP216
株式会社デジテックス研究所
多用途アクティブ電極×12
REF, Eアクティブ電極×1ずつ
RESP×3 呼吸入力
pO2×1 パルスオキシメータ入力
EXT×2 外部入力
PULSE×1 脈波入力
6ch
アクティブ電極 300G その他 10MG
6 ビット
~1000Hz 独立設定可能
日本睡眠学会 PSG 共通フォーマット
色対値表示 良否判定表示
り 400g(バッテリー含まず)
$00W \times 50H \times 160Dmm$
OC7.2V2 次電池リチウムイオン
.0VA 以下
CF メモリーカード
AP-C300
JP-QM71D

## B) ソフトウェア

### B-1) C++言語プログラム

#### 視覚刺激提示ソフトウェア

本研究の予備実験において、自由に図形の色や背景色の変更、点滅周波数や刺激時間を変更しやすく、実際の点滅周期が分かるようにソフトウェアを作成した.

ソフトウェアのソースコードを以下に記述する.

SSVEP.cpp ソースコード

```
#include "stdafx.h"
#include "Form1.h"

using namespace SSVEP;

LARGE_INTEGER nFreq, nBefore, nAfter;
DWORD dwTime;

[STAThreadAttribute]
int main(array<System::String ^> ^args)
{

// コントロールが作成される前に、Windows XP ビジュアル効果を有効にします
Application::EnableVisualStyles();
Application::SetCompatibleTextRenderingDefault(false);

// メイン ウィンドウを作成して、実行します
Application::Run(gcnew Form1());
return 0;
}
```

#### Form1.h ソースコード

```
#include <windows.h>
#include <stdio.h>
#pragma comment(lib, "winmm.lib")
#pragma once
namespace SSVEP {
            using namespace System;
            using namespace System::ComponentModel;
           using namespace System::Collections;
using namespace System::Windows::Forms;
            using namespace System::Data;
            using namespace System::Drawing;
            int s1=600;
            int s2=50;
           /// <summary>
/// Form1 の概要
/// </summary>
            public ref class Form1 : public System::Windows::Forms::Form
           public:
                       Form1 (void)
                                   InitializeComponent();
                                   ///TODO: ここにコンストラクター コードを追加します
           protected
                        /// <summary>
                       /// Summary/
/// 使用中のリソースをすべてクリーンアップします。
/// </summary>
~Form1()
                                    if (components)
                                               delete components;
                                   }
            protected:
            private: System::Windows::Forms::Panel^ panel1;
            private: System::Windows::Forms::PictureBox^ pictureBox1;
                                                                    ファイルを点滅;
            private: System::Windows::Forms::Button^
            private: System::Windows::Forms::Label^
                                                                   label1:
            private: System::Windows::Forms::Label^
                                                                   出力回数;
            private: System::Windows::Forms::Label^
                                                                   出力時間;
           private: System::Windows::Forms::Button^ setting;
private: System::Windows::Forms::TextBox^ textBox2;
private: System::Windows::Forms::TextBox^ textBox1;
private: System::Windows::Forms::CheckBox^ checkBox1;
private: System::Windows::Forms::Label^ label2;
private: System::Windows::Forms::Label^ label3;
private: System::Windows::Forms::Button^ button1;
            private: System::Windows::Forms::Button^ button1;
            private: System::ComponentModel::IContainer components;
           protected:
```

```
protected:
           private:
                         /〈summary〉
/ 必要なデザイナー変数です。
                      /// </summary>
#pragma region Windows Form Designer generated code
                      /// <summary>
/// = summary>
/// デザイナー サポートに必要なメソッドです。このメソッドの内容を
/// コード エディターで変更しないでください。
/// </summary>
                      void InitializeComponent(void)
                                 this->panel1 = (gcnew System::Windows::Forms::Panel());
                                 this->button1 = (gcnew System::Windows::Forms::Button());
this->label3 = (gcnew System::Windows::Forms::Label());
                                 this->label2 = (gcnew System::Windows::Forms::Label());
                                 this->checkBox1 = (gcnew System::Windows::Forms::CheckBox());
this->textBox1 = (gcnew System::Windows::Forms::TextBox());
this->textBox2 = (gcnew System::Windows::Forms::TextBox());
this->setting = (gcnew System::Windows::Forms::Button());
                                 this->出力時間 = (gcnew System::Windows::Forms::Label());
                                 this->出力回数 = (gcnew System::Windows::Forms::Label());
                                 this->label1 = (gcnew System::Windows::Forms::Label());
this->ファイルを点滅 = (gcnew System::Windows::Forms::Button());
                                 this->pictureBox1 = (gcnew
System::Windows::Forms::PictureBox());
                                 this->panel1->SuspendLayout();
(cli::safe_cast<System::ComponentModel::ISupportInitialize^
>(this->pictureBox1))->BeginInit();
                                 this->SuspendLayout();
                                     pane l 1
                                 this->panel1->AutoScroll = true;
                                 this->panel1->BackColor =
System::Drawing::SystemColors::Desktop;
                                 this->panel1->Controls->Add(this->button1);
                                 this->panel1->Controls->Add(this->label3);
                                 this->panel1->Controls->Add(this->label2)
                                 this->panel1->Controls->Add(this->checkBox1);
                                 this->panel1->Controls->Add(this->textBox1);
                                 this->panel1->Controls->Add(this->textBox2);
                                 this->panel1->Controls->Add(this->setting);
                                 this->panel1->Controls->Add (this->出力時間);
this->panel1->Controls->Add (this->出力回数);
this->panel1->Controls->Add (this->label1);
                                 this->panel1->Controls->Add(this->ファイルを点滅);
                                 this->panel1->Controls->Add(this->pictureBox1);
                                 this->panel1->Dock = System::Windows::Forms::DockStyle::Fill;
                                 this->panel1->Location = System::Drawing::Point(0, 0);
this->panel1->Name = L"panel1";
                                 this->panel1->Size = System::Drawing::Size(910, 622);
                                 this->panel1->TabIndex = 1;
                                 this->panel1->Paint += gcnew
System::Windows::Forms::PaintEventHandler(this, &Form1::panel1_Paint);
                                 // button1
                                 this->button1->Location = System::Drawing::Point(709, 28);
this->button1->Name = L"button1";
                                 this->button1->Size = System::Drawing::Size(75, 23);
```

```
this->button1->TabIndex = 13;
                               this->button1->Text = L"set"
                               this->button1->UseVisualStyleBackColor = true;
                               this->button1->Click += gcnew System::EventHandler(this,
&Form1::button1_Click);
                                  label3
                               this->label3->AutoSize = true;
                               this->label3->BackColor =
System::Drawing::SystemColors::ButtonHighlight;
                               this->label3->Location = System::Drawing::Point(494, 56);
this->label3->Name = L"label3";
                               this->label3->Size = System::Drawing::Size(41, 12);
                               this->label3->TabIndex = 12;
                               this->label3->Text = L"label3";
                                  label2
                               this->label2->AutoSize = true;
                               this->label2->BackColor =
System::Drawing::SystemColors::ButtonHighlight;
                               this->label2->Location = System::Drawing::Point(494, 19);
this->label2->Name = L"label2";
                               this->label2->Size = System::Drawing::Size(41, 12);
                               this->label2->TabIndex = 11;
this->label2->Text = L"label2";
                                  checkBox1
                               this->checkBox1->Anchor =
static_cast<System::Windows::Forms::AnchorStyles>((System::Windows::Forms::AnchorStyles::
Top | System::Windows::Forms::AnchorStyles::Right));
                               this->checkBox1->AutoSize = true;
                               this->checkBox1->BackColor =
System::Drawing::SystemColors::Desktop;
                               this->checkBox1->Checked = true;
                               this->checkBox1->CheckState =
System::Windows::Forms::CheckState::Checked;
                               this->checkBox1->ForeColor =
System::Drawing::SystemColors::ButtonHighlight;
                               this->checkBox1->Location = System::Drawing::Point(802, 12);
this->checkBox1->Name = L"checkBox1";
                               this->checkBox1->Size = System::Drawing::Size(96, 16);
                               this->checkBox1->TabIndex = 10;
this->checkBox1->Text = L"表示チェック";
                               this->checkBox1->UseVisualStyleBackColor = false;
                               this->checkBox1->CheckedChanged += gcnew
System::EventHandler(this, &Form1::checkBox1 CheckedChanged);
                                  textBox1
                              this->textBox1->Location = System::Drawing::Point(406, 13);
this->textBox1->Name = L"textBox1";
this->textBox1->Size = System::Drawing::Size(81, 19);
                               this->textBox1->TabIndex = 9;
                               this->textBox1->Text = L"600";
                                  textBox2
                               this->textBox2->Location = System::Drawing::Point(406, 50);
                               this->textBox2->Name = L"textBox2";
                               this->textBox2->Size = System::Drawing::Size(81, 19);
this->textBox2->TabIndex = 8;
                               this->textBox2->Text = L"50";
```

```
setting
                             this->setting->Location = System∷Drawing∷Point(606, 29);
                             this->setting->Name = L"setting";
this->setting->Size = System::Drawing::Size(75, 23);
                             this->setting->TabIndex = 7;
this->setting->Text = L"setting";
this->setting->UseVisualStyleBackColor = true;
                             this->setting->Click += gcnew System::EventHandler(this.
&Form1::setting Click);
                             // 出力時間
                             this->出力時間->AutoSize = true;
                             this->出力時間->BackColor =
System::Drawing::SystemColors::ButtonHighlight;
                             this->出力時間->Location = System::Drawing::Point(330, 50);
this->出力時間->Name = L"出力時間";
                             this->出力時間->Size = System::Drawing::Size(53, 12);
                             this->出力時間->TabIndex = 5;
this->出力時間->Text = L"出力時間";
                                出力回数
                             this->出力回数->AutoSize = true;
this->出力回数->BackColor =
System::Drawing::SystemColors::ButtonHighlight;
                             this->出力回数->Location = System::Drawing::Point(330, 16);
                             this->出力回数->Name = L"出力回数"
                             this->出力回数->Size = System::Drawing::Size(53, 12);
                             this->出力回数->TabIndex = 4;
this->出力回数->Text = L"出力回数";
                                label1
                             this->label1->AutoSize = true;
                             this->label1->Font = (gcnew System::Drawing::Font(L"MS ゴシッ
ク", 20.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
                                       static cast<System::Byte>(128)));
                             this->label1->ForeColor =
System::Drawing::SystemColors::ButtonHighlight;
                             this->label1->Location = System::Drawing::Point(601, 83);
this->label1->Name = L"label1";
                             this->label1->Size = System::Drawing::Size(180, 27);
                             this->label1->TabIndex = 2;
                             this->label1->Text = L"点滅間隔時間";
                                 ファイルを点滅
                             this->ファイルを点滅->Location = System::Drawing::Point(27, 13);
this->ファイルを点滅->Name = L″ファイルを点滅";
                             this→>ファイルを点滅→>Size = System::Drawing::Size(286, 87);
                             this->ファイルを点滅->TabIndex = 1;
this->ファイルを点滅->Text = L"ファイルを点滅";
                             this->ファイルを点滅->UseVisualStyleBackColor = true;
                             this->ファイルを点滅->Click += gcnew System::EventHandler(this.
&Form1::ファイルを表示 Click);
                                pictureBox1
                             this->pictureBox1->Anchor =
System::Windows::Forms::AnchorStyles::None;
                             this->pictureBox1->Location = System::Drawing::Point(332, 222);
                             this->pictureBox1->Name = L"pictureBox1";
```

```
this->pictureBox1->Size = System::Drawing::Size(223, 145);
                          this->pictureBox1->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::Zoom;
                          this->pictureBox1->TabIndex = 0;
                          this->pictureBox1->TabStop = false;
                             Form1
                          this->AutoScaleDimensions = System::Drawing::SizeF(6, 12);
                          this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
                          this->BackColor = System::Drawing::SystemColors::Desktop;
this->ClientSize = System::Drawing::Size(910, 622);
                          this->Controls->Add(this->panel1);
                          this->Font = (gcnew System::Drawing::Font(L"MS ゴシック", 9,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
                          static_cast<System::Byte>(128)));
this->Name = L"Form1";
this->Text = L"SSVEP刺激提示ソフト";
                          this->Load += gcnew System::EventHandler(this,
&Form1::Form1 Load 1);
                          this->panel1->ResumeLayout(false);
                          this->panell->PerformLayout();
(cli::safe_cast<System::ComponentModel::ISupportInitialize^
>(this->pictureBox1))->EndInit();
                          this->ResumeLayout(false);
#pragma endregion
                 private: System::Void ファイルを表示_Click(System::Object sender,
System::EventArgs
                   e) {
        LONGLONG begin;
        LONGLONG freq;
        LONGLONG end;
        ::timeBeginPeriod(1);//タイマーの精度を設定
        int i;
        for (i=0; i < s1; i++) {
                  ::QueryPerformanceCounter( (LARGE INTEGER*)&begin);
                  :QueryPerformanceFrequency((LARGE_INTEGER*)&freq);
                 ::Sleep(s2);
                 :: QueryPerformanceCounter((LARGE INTEGER*)&end);
                 switch(this->pictureBox1->Visible)
                 case false:
                          this->pictureBox1->Visible = true;
                          break;
                 case true:
                          this->pictureBox1->Visible = false;
                          break:
                 double a=0;
                 a=double (end-begin) / freq;
                 label1->Text=a.ToString();
                 Application::DoEvents();
pictureBox1->Image = bmp;
```

```
label2->Text=s1. ToString();
                                   label3->Text=s2. ToString();
private: System::Void checkBox1_CheckedChanged(System::Object^ sender, System::EventArgs^
e) {
                                   if (this->checkBox1->Checked == true)
                                                         this->出力回数->Visible = true;
this->出力時間->Visible = true;
                                                         this->textBox1->Visible = true;
                                                         this->textBox2->Visible = true;
                                                         this->setting->Visible = true;
this->label1->Visible = true;
                                                         this->label2->Visible = true;
this->label3->Visible = true;
                                                         this->button1->Visible = true;
                                   else
                                              this->出力回数->Visible = false;
                                              this->出力時間->Visible = false;
this->textBox1->Visible = false;
this->textBox2->Visible = false;
this->setting->Visible = false;
                                              this->label1->Visible = false;
                                              this->label2->Visible = false;
this->label3->Visible = false;
                                              this->button1->Visible =false;
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e) {
                                   this->pictureBox1->Location = System::Drawing::Point(100, 100);
private: System::Void panel1_Paint(System::Object^ sender,
System::Windows::Forms::PaintEventArgs^ e) {
```

### B-2) MATLAB/Simulinkモデルプログラム

#### SSVEP 検出プログラム

本研究での SSVEP を検出プログラムは Simulink で用意されているブロックと M 言語によって作成したブロックを組み合わせたプログラムである. A では STFT による周波数解析と解析後のデータの抽出を行い, B ではデータの移動平均を行い, C ではそれぞれの電極で SSVEP が出現したか, 出現した電極の数から最終的にどの周波数で SSVEP が検出できたかを出力する.

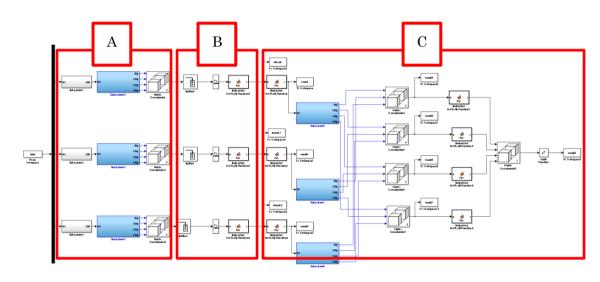


Figure B-2 EEG\_FFT\_SSVEP\_10s\_not8Hz.slx

## B-3) MATLAB M言語プログラム

#### SSVEP 検出プログラムを自動再生させるプログラム

複数の脳波データを自動処理するプログラムを以下に示す.

model\_auto.m ソースコード

```
readdata=readdata(1:10240,4:6);
     elseif finddata(nan(2))>1000
                 nan2(i) = nan(2);
                 readdata=readdata(nan2(i):nan2(i)+10239, 4:6);
     else
            disp('予期しない条件があります。動作を停止します。')
           break
      end
     data=horzcat(timedata.readdata);
     open_system('EEG_FFT_6chSSVEP_10s.slx')
open_system('EEG_FFT_6chSSVEP_10s_not8Hz.slx')
sim('EEG_FFT_6chSSVEP_10s.slx')
sim('EEG_FFT_6chSSVEP_10s_not8Hz.slx')
%
     pause (15)
     \begin{array}{l} \text{data\_save}\,(:,:,1,\,i) \!=\! s\,\text{imout}\,(:,:,41)\,;\\ \text{data\_save}\,(:,:,2,\,i) \!=\! s\,\text{imout1}\,(:,:,41)\,;\\ \text{data\_save}\,(:,:,3,\,i) \!=\! s\,\text{imout2}\,(:,:,41)\,; \end{array}
                                                               %sheeti番目の電極1番
                                                               %sheeti番目の電極2番
                                                               %sheeti番目の電極3番
      lastresult(:, :, i)=result3(:, :, 41);
end
run ('excel template.m')
run('border_excel.m')
      STFT, 移動平均後の結果を MS-Excel に出力するプログラム
      電極毎のSTFT、移動平均後の結果をMS-Excel に出力するプログラムを以下に示す。
      Excel template.m ソースコード
transport range=100;
datacontainer=zeros(transport_range, transport_range);
transport_elctrode_number=3;
datarowsize=33;
% data electrode=zeros(33, 4, 3, 12);
data_electrode=zeros(33, 3, 3, 12);
electrode number=1;
sheetnumber=7;
for shenum=1:sheetnumber
     \label{lem:data_electrode} $$ \begin{array}{l} \text{data\_electrode}(:,:,1,\text{shenum}) = \text{data\_save}(:,:,1,\text{shenum}) ; \\ \text{data\_electrode}(:,:,2,\text{shenum}) = \text{data\_save}(:,:,2,\text{shenum}) ; \\ \text{data\_electrode}(:,:,3,\text{shenum}) = \text{data\_save}(:,:,3,\text{shenum}) ; \\ \end{array} $$
           for j=1:transport_range
                  if rem(j, 6) == \overline{1}
                       datacontainer(1, j) = electrode_number;
                       for i=2:1+datarowsize
                             datacontainer (i, j) = 2.048 + 0.256 * (i-2);
                       end
                       for x=1:datarowsize
                             for y=1:3
                                                                       %最大値は取り出す周波数の数
```

#### 最終的な SSVEP 検出結果を MS-Excel に出力するプログラム

end

最終的にどの時間でどの周波数の SSVEP が検出できたのかの結果を MS-Excel に出力するプログラムを以下に示す.

```
Border_excel.m

sheetnumber=7;

for i=1:32
    timedata1(i,1)=2.304+0.256*(i-1);
end

for shenum=1:sheetnumber
    datacontainer=lastresult(:,:,shenum);
    datacontainer=horzcat(timedata1,datacontainer);

xlswrite('F:¥修士研究¥実験データ¥2016-01-27¥15Hz400青白¥15Hz注目¥閾値100適用15Hz注目not8Hz.xlsx',datacontainer,shenum);
```

### B-4) MS-Excel Visual Basicプログラム

#### グラフ作成プログラム

MATLAB M 言語「STFT、移動平均後の結果を MS-Excel に出力するプログラム」から出力された Excel ファイルのデータをグラフ化するプログラムを以下に示す.

#### Sub makegraph()

```
Sub makegraph()
```

Dim lngYLine As Long Dim intXLine As Integer Dim objFind As Object Dim strAddress As String Dim i As Integer Dim sheet1 As Worksheet Dim YScale(12) As Integer

For x = 1 To 7

Set sheet1 = Worksheets(x) sheet1.Activate アクティブにする

アクティブにする Set objFind = sheet1.Cells.Find("2.048", LookAt≔xlWhole) 'sheet1 内から 2.048 を検索する

If Not objFind Is Nothing Then strAddress = objFind.Address 'もし検索結果が Nothing でないなら

'sheet1 に代入した Worksheet 番号を

'Worksheetx 番目を代入

Do While Not objFind Is Nothing lngYLine = objFind.Cells.Row intXLine = objFind.Cells.Column

'もし検索結果が Nothing でないなら

 $\begin{array}{l} Cells(lngYLine - 1, intXLine + 1) = "10Hz" \\ Cells(lngYLine - 1, intXLine + 2) = "12Hz" \\ Cells(lngYLine - 1, intXLine + 3) = "15Hz" \\ Columns(intXLine + 4).Clear \\ Columns(intXLine + 5).Clear \end{array}$ 

Set rng\_Max = Range(Cells(lngYLine, intXLine + 1), Cells(lngYLine + 33, intXLine + 3))

maxdata\_new = WorksheetFunction.Max(rng\_Max)

If maxdata\_new > maxdata\_old Then

maxdata\_old = maxdata\_new

End If

 $\begin{array}{l} maxzyozan = maxdata\_old ~ \S ~ 500 \\ YScale(x) = (maxzyozan + 2) * 500 \end{array}$ 

Set objFind = Cells.FindNext(objFind)

If strAddress = objFind.Address Then Range(Columns(intXLine + 6), Columns(intXLine + 100)).Clear Range(Rows(lngYLine + 33), Rows(lngYLine + 100)).Clear maxdata\_new = 0

```
maxdata old = 0
               Exit Do
         End If
     Loop
Else
     MsgBox "ご指定の範囲に起点データが存在しません"
End If
maxdata_new = 0
maxdata\_old = 0
Next
For j = 1 To 7
Set sheet1 = Worksheets(j)
sheet1.Activate
                                                     'Worksheet1 番目を代入
'sheet1 に代入した Worksheet 番号を
アクティブにする
ActiveWindow.Zoom = 100
                                                        'sheet1 のウィンドウ縮尺を10
0%にする
Set objFind = sheet1.Cells.Find("2.048", LookAt:=xlWhole) 'sheet1 内から 2.048
を検索する
If Not objFind Is Nothing Then strAddress = objFind.Address
                                                     'もし検索結果が Nothing でないなら
     Do While Not objFind Is Nothing lngYLine = objFind.Cells.Row
                                                     'もし検索結果が Nothing でないなら
          intXLine = objFind.Cells.Column
          i = i + 1
With sheet1.ChartObjects.Add(54 + 326 * (i - 1), 500, 324, 200).Chart .ChartType = xlXYScatterSmooth .SetSourceData Source:=Range(Cells(lngYLine - 1, intXLine), Cells(lngYLine - 1 + 33, intXLine + 3)) .Axes(xlValue).MaximumScale = YScale(j)
          .Axes(xlCategory).HasMajorGridlines = True
End With
          Set objFind = Cells.FindNext(objFind)
         If strAddress = objFind.Address Then
        Exit Do
End If
     Loop
     MsgBox "ご指定の範囲に起点データが存在しません"
End If
i = 0
ActiveWindow.Zoom = 70
                                                       'sheet1のウィンドウ縮尺を70%
Next
                                                     'Worksheet1 番目を代入
'sheet1 に代入した Worksheet 番号を
Set sheet1 = Worksheets(1)
sheet1.Activate
アクティブにする
```

End Sub