

行政院及所屬各機關出國報告
(出國類別：進修)

腦磁圖之操作與臨床運用

服務機關：台北榮民總醫院
出國人職稱：住院醫師
姓名：尤香玉

出國地區：日本仙台
出國期間：90.7.1. — 90.12.31.
報告日期：91.年3.月5.日

J3/
09003205

系統識別號:C09003205

公 務 出 國 報 告 提 要

頁數: 17 含附件: 否

報告名稱:

腦磁圖之操作與臨床運用

主辦機關:

行政院輔導會臺北榮民總醫院

聯絡人/電話:

/

出國人員:

尤香玉 行政院輔導會臺北榮民總醫院 神經醫學中心 住院醫院

出國類別: 進修

出國地區: 日本

出國期間: 民國 90 年 07 月 01 日 -民國 90 年 12 月 31 日

報告日期: 民國 91 年 03 月 05 日

分類號/目: J3/醫療 J3/醫療

關鍵詞: 腦磁圖之操作與臨床運用

內容摘要: 摘要腦磁圖 (Magnetoencephalography) 為一利用超低溫導體來偵測及記錄生物腦部訊息的磁場，近年來已漸漸應用於腦功能之研究、臨床合併腦電波解讀生理訊號。各國醫學中心陸續導入腦磁圖 (Magnetoencephalography) 系統，搭配正子發射紀錄儀 (Positron Emission Tomography)，功能性核磁共振儀 (Function MRI)，作為臨床研究神經生理及腦功能的重要工具。由於本院於 2001 年裝置了腦磁圖，有此人力需要，便前往腦磁圖裝置最多的日本進修腦磁圖之操作與臨床運用。進修地點為日本仙台東北大學附屬醫院暨廣南病院，該院之腦磁圖以應用在臨床多年，經驗相當豐富。由 2001 年七月份開始為期半年，初期以熟悉腦磁圖系統的操作，並參與臨床癲癇病人之錄影及腦硬膜下電極板之腦波紀錄及刺激，中期便實際病人腦磁圖之計測和資料分析，並作成書面報告。後一個月為論文寫作，計畫投稿。此次進修心得除已熟悉腦磁圖之操作，並學習到神經生理訊號判讀之技巧，要充分運用腦磁圖。使用者的神經生理背景及電腦知識是不可或缺的。團隊的合作及實驗室使用準則的確立更可使整個團隊的研究成果提升。

本文電子檔已上傳至出國報告資訊網

摘要

腦磁圖 (Magnetoencephalography) 為一利用超低溫導體來偵測及記錄生物腦部訊息的磁場，近年來已漸漸應用於腦功能之研究：臨床合併腦電波解讀生理訊號。各國醫學中心陸續導入腦磁圖 (Magnetoencephalography) 系統，搭配正子發射紀錄儀 (Positron Emission Tomography)，功能性核磁共振儀 (Function MRI)，作為臨床研究神經生理及腦功能的重要工具。由於本院於 2001 年裝置了腦磁圖，有此人力需要，便前往腦磁圖裝置最多的日本進修腦磁圖之操作與臨床運用。

進修地點為日本仙台東北大學附屬醫院暨廣南病院，該院之腦磁圖以應用在臨床多年，經驗相當豐富。由 2001 年七月份開始為期半年，初期以熟悉腦磁圖系統的操作，並參與臨床癲癇病人之錄影及腦硬膜下電極板之腦波紀錄及刺激，中期便實際病人腦磁圖之計測和資料分析，並作成書面報告。後一個月為論文寫作，計畫投稿。此次進修心得除已熟悉腦磁圖之操作，並學習到神經生理訊號判讀之技巧，要充分運用腦磁圖。使用者的神經生理背景及電腦知識是不可或缺的。團隊的合作及實驗室使用準則的確立更可使整個團隊的研究成果提升。

目次：

一、 目的

二、 過程

三、 心得與建議

四、 附錄

一、 目的:

腦磁圖 (Magnetoencephalography) 為一利用超低溫導體來偵測及記錄生物腦部訊息的磁場，近年來已漸漸應用於腦功能之研究、臨床合併腦電波解讀生理訊號。各國醫學中心陸續導入腦磁圖 (Magnetoencephalography)系統，搭配正子發射紀錄儀(Positron Emission Tomography)，功能性核磁共振儀(Function MRI)，作為臨床研究神經生理及腦功能的重要工具。由於造價昂貴，目前腦磁圖設備多集中於已開發國家如美國、北歐、日本等地區。其中日本的數量最多至 2001 年已有 22 套腦磁圖裝置於各大醫院，近十年關於腦磁圖的研究除北歐佔領導地位外，屬日本的研究人數及研究數量最多。日本東北大學附屬醫院暨廣南病院為日本最早期使用腦磁圖的機構，將腦磁圖運用於臨床至少十年，負責腦磁圖的中里信和醫師為首位發表利用腦磁圖記錄到人類阿爾發波並發表成論文。本院於 2001 年引進腦磁圖系統並與正子發射紀錄儀(Positron Emission Tomography)，功能性核磁共振儀(Function MRI)整合成腦功能整合研究中心。由於人力需要，日本又與本國距離相近，於是前往日本東北大學進修腦磁圖之操作與應用。

二、 過程：

2001年六月底以請休假方式先抵達日本仙台，住宿地點決定後，於七月二日報到，並拜訪前東北大學附屬醫院醫院院長及神經外科教授吉本高志先生。次日正式開始進修，七月份以熟悉整個作業系統為主，電腦連線設定及不同語言版本之電腦作業系統之整合，並開始計畫研究之方向。臨床部分則參與癲癇手術病患之術前評估包括硬腦膜下腦電波紀錄及刺激。除每日晨會外，每週有一次期刊討論會，每兩週有病例討論會，學術活動頗多。

八月初由於廣南病院的腦磁圖系統更新及日本當地的暑假，無法使用。本人於八月十日回台一週，並更換簽證為留學簽證，若不更換只能在當地停留九十日。九月奉教授指派至日本青森參加日本腦神經外科學會，發表口頭報告『右腦半球為語言中樞之癲癇病人硬腦膜下腦電波紀錄及刺激』。廣南病院之腦磁圖雖健保不幾付，但病人可自費，每次計價為十萬日幣，每日約有兩位病人的計測。必須於一週內作成書面報告並存檔。由於在台北榮總神經內科已有打腦波報告的經驗，腦磁圖報告很快就上手，九月底至十一月初本人便負責所有的計測及報告。十一月初日本神經生理學會年會於東京舉行，本人亦前往參加，會中邀請知名國際學者發表演講，包括腦磁圖的分野，另外經頭顱磁刺激的部分也相當豐富。晚上有一腦磁圖使用者交流會，由儀器製造商Neuromag舉辦，邀請國際有名專門研究Mismatch Negativity的學者(Dr. Marie)作專題演講，十分精采，另外目前腦磁圖在臨床上如何可被廣泛應用及各國擬定腦磁圖的保險幾付也在會中探討。與會期間本人受東京都立府中醫院神經外科清水教授及川合醫師之邀參觀該院，並於手術室中觀摩腦樑離段手術。清水教授及川合醫師曾多次到台北榮總參觀，與本科醫師多有交流，此次參觀該院對於癲癇手術部分的交流幫助

甚大。十一月份也開始論文的寫作與準備，由於本人之前所學多與癲癇及腦電波相關，與中里醫師討論結果決定將重點放在癲癇病人腦磁圖研究，收集資料後研究的課題為『兩側同期性癲癇棘波於腦磁圖上的表現』十二月份開始著手撰寫，並繼續計測及報告之工作，於十二月底大致完成但仍在修改階段。日本於十二月二十四日便開始放年假(天皇生日連假)，元旦兩天假又碰到週六週日，一直到一月六日才正式恢復上班。本人沒有經驗，於是將住宿房間退還及停用電話、網路等手續無法辦理。回國行李託運、稱重也無法進行加上論文繼續修改，所以延至一月十日才返抵台北，並於當日下午銷假上班。

三、 心得與建議;

此次為期六個月之進修主要收穫為對於 Neuromag 腦磁圖系統操作熟悉；對於腦磁圖究竟在研究領域及臨床能夠做什麼更加了解。以往神經內科內常用的神經生理訊號包括誘發電位及腦波都可以用腦磁圖加以擴充，對於腦部不同部位功能的定位更有幫助；在癲癇部分腦磁圖對於腦溝發出之棘波更敏感，能補現行腦電波之不足。腦磁圖所得之磁場訊號必須由電腦軟體再處理運算，以電流雙極子的概念呈現，所以腦磁圖最終的結果多半是一以電腦運算過的資料表示，對於這樣的結果，判讀者必須再過濾，自己判斷及解釋最後的意義，不能一味接受電腦運算的結果。所以關於神經生理訊號的判讀，無論機器多先進，判讀者本身對神經生理的基礎概念及經驗更重要。

現今神經科學的發展一為遺傳、分子生物學方面，另一大方向就是經由神經生理訊號來研究人腦的功能，在這方面數位化的訊息處理為必備的工具，從事這方面的研究必須對電腦相當熟悉，這其中包括檔案的交換、傳輸及常用軟體的熟悉。對於新開發之軟硬體都要能快速上手。在廣南病院的腦磁圖室就有 Neuromag 工程師常駐，當有電腦及技術上的問題馬上就能的到協助。在本院則每一位腦磁圖使用者自己對於整個系統要有某種程度的熟悉，才能獨立作業。建議有志參與腦磁圖的人員對於常用的軟體及基本電腦的嘗試要具備。

腦磁圖要廣泛應用必先建立其可信度。侵入性的檢查如手術中大腦功能定位及腦棘波紀錄都可用來驗證腦磁圖及功能性核磁共振的可用性。本院腦神經外科在這方面已相當有經驗若能與腦磁圖部分合作相信會有更多研究素材及成果。

附錄：

附錄 1：研究論文大綱

附錄 2：腦磁圖軟體攻略

Neuromagnetic Separation of Secondary Bilateral Synchronized Spikes

*Hsiang-yu Yu , Nobukazu Nakasato, Hiroshi Shamoto , Masaki Iwasaki,
Ken-ichi Nagamatsu, Takashi Yoshimoto

Department of Neurosurgery, Tohoku University School of Medicine and Kohnan
Hospital

*Neurology Department, Taipei Veterans General Hospital

Purpose

- To demonstrate the spatiotemporal advantage of MEG we present:
 - two cases with synchronized spikes at mirror foci, and
 - one case of frontal lobe epilepsy with secondary bilateral synchrony

Case 1

- 29 y/o female, right handed
- Sz onset :12 y/o
- Complex partial seizures
- EEG: synchronized bilateral centrotemporal spikes
- SPECT: interictal hypoperfusion, right temporal
- MRI: left mesial temporal atrophy

MEG/EEG

Case 2

- 20 y/o male, right handed
- Sz onset: 12 y/o
- Sz pattern: left facial and hand twitching
- EEG: bilateral synchronized centrotemporal spikes.
- MRI: normal
- Subdural grid implantation and focal resection

MEG/EEG Grid Case 3

- 14 y/o female , right handed
- Sz patterns: staring, vocalization and GTCs
- EEG: 2.5-3Hz generalized spike-and-wave complexes, left prominent
- MRI: normal
- Subdural grid study, no resection

MEG/EEG Grid Discussion

- Interhemispheric transmission to a mirror focus can be well demonstrated by MEG.
- Latencies around 20ms suggested callosal transmission
- Frontal lobe epilepsy with secondary bilateral synchrony can also be demonstrated on MEG.
- Spike propagation of FLE with SBS could be related to a central cephalic spreading

(cd)

unix の command: change directory

(add-button *user-menu* "ABC")(cd)

User に ABC という button を加え、その機能を cd とする。

応用例

```
(setq open-diskfile-button
      (add-button *file-menu* "Open Diskfile" '(open-diskfile :interactive)))
```

(add-separator)=(make-separator)

似たようなものです。

(open-diskfile :interactive)

diskfile を開く

(close-diskfile :interactive)

diskfile を閉じる

(add-separator *user-menu*)

separator を入れる。

(manage *control-panel*)

control panel を表示させる。

(load-settings :interactive)

setup file を開く。

(save-settings :interactive)

setup file を保存する。但し string の引数を必要とする。

(str-append "aiu" "eo")→"aiueo"

string を足す。

(getenv "HOME")→"/net/minea/home/Neurosurgery"

HOME directory の directory 名を表示させる。

その他 LISPROOT USERROOT

(user-debugger)

signal processor に Debug(1)などという program が走る。

(logo "ABC")

logo が表示される。string の引数が必須。

(version-number-string)

Graph の version が示される。

(export *...*)

local symbol を package に認識させる。

package とは lisp symbol を集めたもの

local symbol と external symbol があるが、local symbol は export して認識させる必要がある。package は通常 user に保存されていて、それが current package となっている。

(manage)

X-library の XtManageChild() と同じもの。X-widget が見えるようになる。

WARNING constant already defined:

**** SIGNAL CATCHED :Signal SIGSEGV (11: segmentation violation)**

Memory allocated (2.59 Mb)

Shall we make coredump (y/n)?

プログラムが壊れ、以後 X-widget の操作を行うのは危険。不適切な引数が X-library に入った可能性がある。Graph は X-library に入る引数は check しない。

(in-package)

current package を設定する。

(load* <functionname>)

lisp の機能を表示させる。

(describe '...)

lisp 内の変数の機能を表示させる。

(require-widget 'pick 'meg)

pick-widget "meg" が作られる。

(cond

((最初の判定条件 最初の掃結動作)
(2番目の判定条件 2番目の掃結動作)
(3番目の判定条件 3番目の掃結動作)
...)

例

```
(defun triangle-using-cond (number)
  (cond ((<= number 0) 0)
        ((= number 1) 1)
        (> number 1)
        (+ number (triangle-using-cond (1- number))))
)
```

(triangle-using-cond 0)→0

(triangle-using-cond 1)→1

(triangle-using-cond 5)→1 原因不明

(matrix-element-sum #m((1 3)(4 5)))→13

(add-path "neuro/lisp/local/code")

path を追加する。

(libraries *graph-root* "command" ...)

graph-root に以下の directory を含ませる。

(libraries *lisp-root* "sys" ...)

lisp-root に以下の directory を含ませる。

(file-exists-p "A")

A という file があれば t。

(filename-as-directory *setup-file-path*)

setup-file-path を表示させる。

(warn "ABC")

signal processor に WARNING "ABC" と表示させる。

(max 1 2 3 2 0)→3

最大値を出す

(min 1 2 3 2 0)→0

最小値を出す

??(x-to-sample widget start)

widget の smp の開始点を出す

その他 (x-to-sample widget end)

(x-start widget)

widget の sec の開始点をだす。

例 (x-start (G-widget "display"))→0.00000E+00

(x-end widget)

widget の sec の終了点をだす。

(define-parameter-group 'rejection "Parameters of std-reject operation")

Parameters に rejection の button を加える。

(defuserparameters rejection

(excluded-meg-chs nil : "Excluded meg-channels" 'list))

rejection に excluded-meg-chs の項目を設け、list が入るようにする。default は nil。

(help-on-version)

Graph、lisp の version が表示される。

全配列の合計を示す。

(setq ABC (random-matrix 10 20))
10x20 の乱数行列 ABC を作る。

<<batch-processor の代わり>>

```
(defun ABC()
;name: /home/Neurosurgery/lesson1.lsp
  (gc)
  (set-resource (G-widget "M-fil") :start -0.05 :end 0.4)
  (average 10 20)
  (average 21 30 :append t)
  (save-average "/neuro/data/ns/case_0148/010613/SEF")
)
```

として

```
(require "/home/Neurosurgery/lesson1")
```

(ABC)

とすれば加算・名前の保存ができる。

(print-array 配列 or 行列)

Signal processor に #m(...)...(...) が表示される。

(setq *print-array* t)

配列が全て表示されるようになる。

print-length 初期値 50

print-depth 初期値 10

*print-radix: 基数

print-base: the base of printed number.

(require-widget 'pick "meg")

meg という名前の pick-widget が作成される。

その他

```
(require-widget 'fft-filter 'M-fil)
```

```
(require-widget 'matrix-source 'ABC)
```

(set-resource (G-widget "filter") :size 1024)

filter の size が 1024 となる。

その他

```
(set-resource (G-widget "meg") :names '(MEG*))
```

```
=(set-resource (G-widget "meg") :names ("MEG*"))
```

→MEG* が選ばれる。

```
(set-resource (G-widget "meg1") :names ("MEG[112 113 122 123]"))
```

→MEG[112 113 122 123] が選ばれる。

```
(set-resource (G-widget "display") :length 6 :increment 30 :default-color 'green)
```

→display の length が 6 秒、increment が 30%、線の色が green になる。

`(set-resource (G-widget "meg") :face-x 0 :face-y 0 :visibility 1)`
→コントロールパネルの(0,0)に meg が移動し、pop-up menu が表示される。

`(setq average-rejection 'std-reject)→std-reject`
average-rejection が std-reject となる。
その他

`(setq average-rejection 'nil)=(setq average-rejection nil)`

`(link (G-widget 'A) (G-widget 'B))`
A から B へ link する。

`(open-diskfile "/neuro/data/ns/AAA/010619/epilepsy")`
`(open-diskfile :interactive)`
disk を開く。

データを切り出して1つのfileにするまでの課程

`(setq AAA (get-data-matrix (G-widget "buffer") 0 600)→#<234x600 matrix>`
`(setq BBB (get-data-matrix (G-widget "buffer") 900 300)→#<234x300 matrix>`
`(setq CCC (mat-append AAA BBB))→#<234 × 900 matrix>`
`(require-widget 'matrix-source 'ABC)`
`(set-resource (G-widget 'ABC) :matrix CCC)`
`(link (G-widget "file")(G-widget "ABC)ここで coil 位置、周波数が継承される。`
`(make-fiff-file :filename "/neuro/data/ns/case_0145/010620/append.fif" :source (G-widget`
`"ABC") :start 0 :end 1.5)`
これで 0~1 秒、1.5~2 秒間の data 1.5 秒分が append.fif として保存される。

`(sample-to-x (G-widget "display") 10)→1.66496E-02`
sample 数を秒に変換する

`(x-to-sample (G-widget "display") 1)→601`
秒をサンプル番号に変換する。

`(x-selection)→(1.21009E+01 4.38717E+00 #<plotter: display>)`
ハイライト部分の開始秒、長さを表示する。
`(setq AAA (x-selection))`
`(car AAA)→1.21009E+01 : 開始秒表示`
`(cadr AAA)→4.38717E+00 : 長さ表示`
`(+ (car AAA)(cadr AAA)) : 終了時間表示`

注 : lisp command の car cadr cdr cons

`(setq AAA '(1 2 3 4 5))`
`(car AAA)→1`
`(cadr AAA)→2`
`(cdr AAA)→(2 3 4 5)`
`(cons 0 AAA)→(0 1 2 3 4 5)`

(manage *control-panel*)

control panel が開く

(defvar ABC 5 "This defines ABC as 5")

ABC を定数 5 と定義する。

(link nil-pointer (G-widget 'A))

A に入ってくる link を消す。

例

(require-widget 'ringbuffer 'buf)

(link (G-widget 'file)(G-widget 'buf))

(link nil-pointer (G-widget 'buf))

(link (G-widget 'A) nil-pointer)は不可

関数の作り方

(defun 関数名(引数 引数 ...)

"説明"

実行文

実行分

)

例

(defun ABC(a b)

"I love you"

(setq AA (* a 3))

(setq BB (* b 4))

)

(ABC 3 4)

AA→9、BB→16

(str-append "ABC" "DEF")→"ABCDEF"

(eql (* 2 2) 4)→t

(G-widget 'file)→#<diskfile :file>

G-widget の名前、種類を表示させる。

(warn "Go to the hell")

Signal processor に WARNING: Go to the hell.

(define-parameter-group 'ABC "abc")

*parameters-menu*に ABC という項目が作られ、ABC→abc と表示される。

さらに

(defuserparameters ABC (AAA 1 "aaa" 'number))
とすると ABC に AAA という項目と 1 という数字が定義される。

(getenv "GRAPHROOT")→"/neuro/lisp/graphV2.92"
(getenv "LISPROOT")→"/neuro/lisp/lispV2.35"
(getenv "USERROOT")→nil

(setq ABC (make-matrix 10 10 1.5))
→#m((1.5 1.5 ... 1.5)(1.5 1.5 ... 1.5)...(1.5 1.5...1.5))が作られる。

(setq ABC (random-matrix 10 10))
→#m(乱数)乱数行列が作られる。

(row ABC)
行列の行が表示される。