

## 認知ゲーム実験 (5) 認知地図の整列効果

山 上 暁

Cognitive game experiment (5) : Alignment effect in cognitive map

YAMAGAMI Akira

**Abstract :** As the fifth report on the cognitive game experiments, I introduce here a game experiment of the alignment effect in cognitive map. The game situations were produced on a Windows XP PC with a program of the HSP language. Players were presented a five-segment route on a city-block map in one of four possible orientations (0, 90, 180, 270 degrees counter-clockwise) for three seconds. After it disappeared, the start point and the first segment of the city-block map were presented to reproduce the memorized route by clicking the line segments on the map. In the “aligned conditions” the orientation of the presented route map and the reproducing map were identical, whereas the “nonaligned conditions” they were different. The results of the completion time and the correct response of the segment clicking showed a clear “alignment effect” in the cognitive map. In the aligned conditions players took shorter time to complete and their responses were more correct than nonaligned conditions.

**Key words :** cognitive game experiment, cognitive map, alignment effect

### 1. 環境心理学と認知地図

環境心理学 (environmental psychology) は人間と環境をひとつの系 (system) として捉えた場合の心理学的諸側面を研究する実証科学としての心理学の一領域であり、環境の知覚・認知、環境評価、環境と社会的プロセス、環境アセスメントとデザインなどの研究領域を含む (羽生, 2006)。人が環境を認知し、その中で行動する場合の環境に関する知識と空間表象を「認知地図」(cognitive map) と呼ぶ。認知地図という心理学的用語は新行動主義の提唱者であるトールマンによって1948年の論文で最初に使用された。ネズミが迷路の学習で報酬である餌の空間位置を迷路の個々のつながりとしてではなく、出発点と目的の位置という「手段-目的関係の認知構造」の中で空間位置関係を学習したという考え方から使われた (森・井上・松井, 1995)。この概念が人間の空間行動に適用されたのは

リンチの1960年の「都市のイメージ」の研究であり、彼は都市の認知地図の構成要素として、パス (path, 経路) ・エッジ (edge, 縁) ・ノード (node, 結節点) ・ディストリクト (district, 領域) ・ランドマーク (landmark, 目印) の5要素を区別し、都市のイメージの分かりやすさがこれらの要素の関係によって規定されていることを示した (ギフォード, 2005)。認知地図は行動地理学においては「個人の日常的な空間環境における現象の相対的位置に関する情報の獲得・コード化・貯蔵・想起・コード解読という一連の心理的変換過程としての認知地図形成の産出物」として定義されている (若林, 1999)。比喻としての「頭の中の地図」やその表現形としての手描きマップについてはこれまでの心理学的・地理学的の研究の中で「イメージ・マップ」(加藤, 1986) や「スケッチ・マップ」(横山, 1997), 「メンタル・マップ」(グールド・ホワイト, 1981) などの用語が用いられてきている。認知地図は空間的知識が記憶の中で空間領域ごとに分節化された

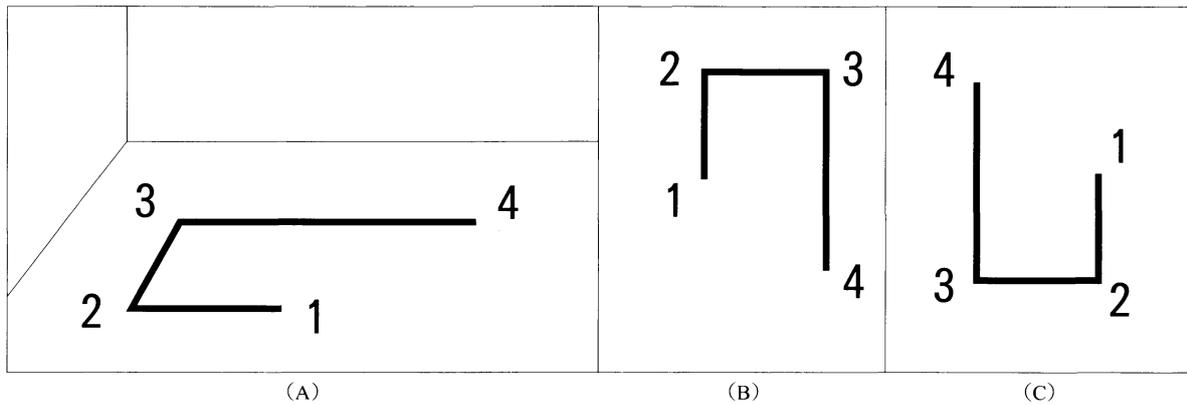


図1 認知地図の整列効果の実験で使用された床上のパスと2種類のマップの例

階層的構造を持って組織化されている一種のネットワークモデルと考えられる場合も多く、またその形成過程に対する情報処理モデル（入力 - 符号化 - 貯蔵 - 解読 - 再生）という形で捉えられるようになってきている（若林，1999）。

頭の中の地図である認知地図は2種類に分けられる。通いなれた最寄の駅から大学までのルートを思い浮かべて何処にどんな店があるかを説明することができる。このような道筋としての認知地図はルートマップと呼ばれる。長く通っている大学ならば、そのキャンスマップを描くこともできる。このタイプの鳥瞰図的な認知地図はサーベイマップと呼ばれる（福田・渡辺，1996）。これらの区分はまずその入力符号化過程あるいは形成過程における直接経験と間接経験（若林，1999）の区分と対応していると考えられる。自分で歩いて通う道は直接経験的であり、ルートマップを形成していくことにつながり、記憶においてはエピソード記憶の系列となる。他方はじめて大学案内で見たキャンスマップは間接経験的でありサーベイマップの形成が意味記憶としてなされる。

認知地図の大きな特徴はその「不正確さ」であり、特有の歪みが距離・角度・方向・面積などの次元で見られる。例えば、距離は小さい距離は過大視され、大きい距離は過小視される傾向や途中に多くの対象物がある距離は過大視される傾向がある。角度においては直角が選好され、小さい角度は過大視され、大きい角度は過小視される傾向がある（森・井上・松井，1995）。このような「歪み」は単なる誤りではなく、人が外界の情報を処理するさいの固有の傾向であり、人間の空間認知の性質を知る手がかりとなる（若林，1999）。認知地図のもうひとつの特徴は「上下左右」の向きあるいは「東西南北」の方向に関して等質ではないこと（異方性）である。これは次の「認知地図の整列効果」

において詳しく述べよう。

## 2. 認知地図の整列効果

地図を見て場所や方向を見つけるとき、地図の方向と実際の環境の方向が一致している方が一致していない場合よりわかりやすい。これを地図の「整列効果」という。認知地図にもこのような整列効果があることを Levin, Jankovic & Palij (1982) は図1に例示したようなパスマップを使用した実験で示した。図の(A)は床の上のパスで、(B)と(C)は方向を変えた2枚のマップである。学習条件は2種類あり、同時的視覚的学習条件では被験者は(B)あるいは(C)のマップを目で見て覚え、継時的運動学習条件ではそのマップを目隠して指でなぞって覚える。そのあと被験者は目隠しをして(A)のように床の上にパスが描かれた実験室でパス上のどの地点にどちらを向いて立っているかを伝えられたあと、指示された地点に歩いて移動するように求められた。例えば「地点4に立って、地点3が前（あるいは後ろ）にあるので、そこから出発して地点1に向かってください。」という形になる。学習時のマップの方向と歩行課題時のパスの方向が一致している場合が整列条件であり、反対方向である場合が反整列条件となる。実験の結果からはどちらの学習条件でも整列条件の方が反整列条件よりも課題遂行成績がよいという「整列効果」が見られた。

以下に示す実験ではPCディスプレイ上のグリッド状のマップの上に5直線セグメントからなるルートを提示し、3秒後に再生する課題を一種のパズルゲームの形で行わせて、そのような条件下での「整列効果」の有無と性質を明らかにするためにルート完成までの時間とセグメント選択の正答スコアを分析した。各試行でのスコアが試行の最後に表示された。

### 3. 整列効果の認知ゲーム実験

#### 目的

認知地図における整列効果を PC ディスプレイ上に作成したグリッド状のマップの上に示されたルートを用いて検討する。記憶した時と同じ方向で再生する場合を整列条件とし、異なる場合を非整列条件とする。整列効果はゲーム実験の誤答と反応時間のデータから分析する。仮説は「記憶ルートと反応ルートの方向が同じである場合（整列条件）より、異なる場合（非整列条件）の方に誤反応が多く、かつ反応時間も長くなる。」とした。

#### 方法

実験参加者は甲南女子大学学生36名。学内のパソコン教室でそれぞれの学生が各自の前の PC に向かい、複数の学生が一斉に自分のペースで実験を行った。実験装置としては Windows PC を用い、HSP プログラム言語による実験プログラムを使用して刺激図形（ルー

トマップ）の提示を個人ごとに異なるランダムな順で行ない、マウスによる反応の入力を記録した。刺激図形は基本ルート3図形（図2の番号1, 5, 9の図形）とそれをそれぞれ反時計周りに90度, 180度, 270度回転させた4方向の図形で合計12枚の図形を作成した。さらに反応用の図形としてオリジナル図形（0度）の方向で出発点を示した図形（A）それを180度回転させた図形（B）の2種類が用意し、それらを全て組み合わせて3図形×4方向×2反応=24刺激を作成した。

実験課題はランダムに提示された24枚の図形を記憶してマウスクリックで記憶した通りに直線セグメントを4本選択する再生課題であった。それぞれのセグメントを選んだときの反応時間とその順序が記録された。実験計画は3要因のデザインで、3図形×4提示方向×2反応方向の全24試行を一人の被験者がすべて行う被験者内計画であった。

ゲームの方法として次のような教示を PC 画面上に示した。「ルートを覚える実験です。はじめに出た画面のルートを覚えてあとの画面でそれを再現してください。ルートを再現する時、出発点と最初のセグメン

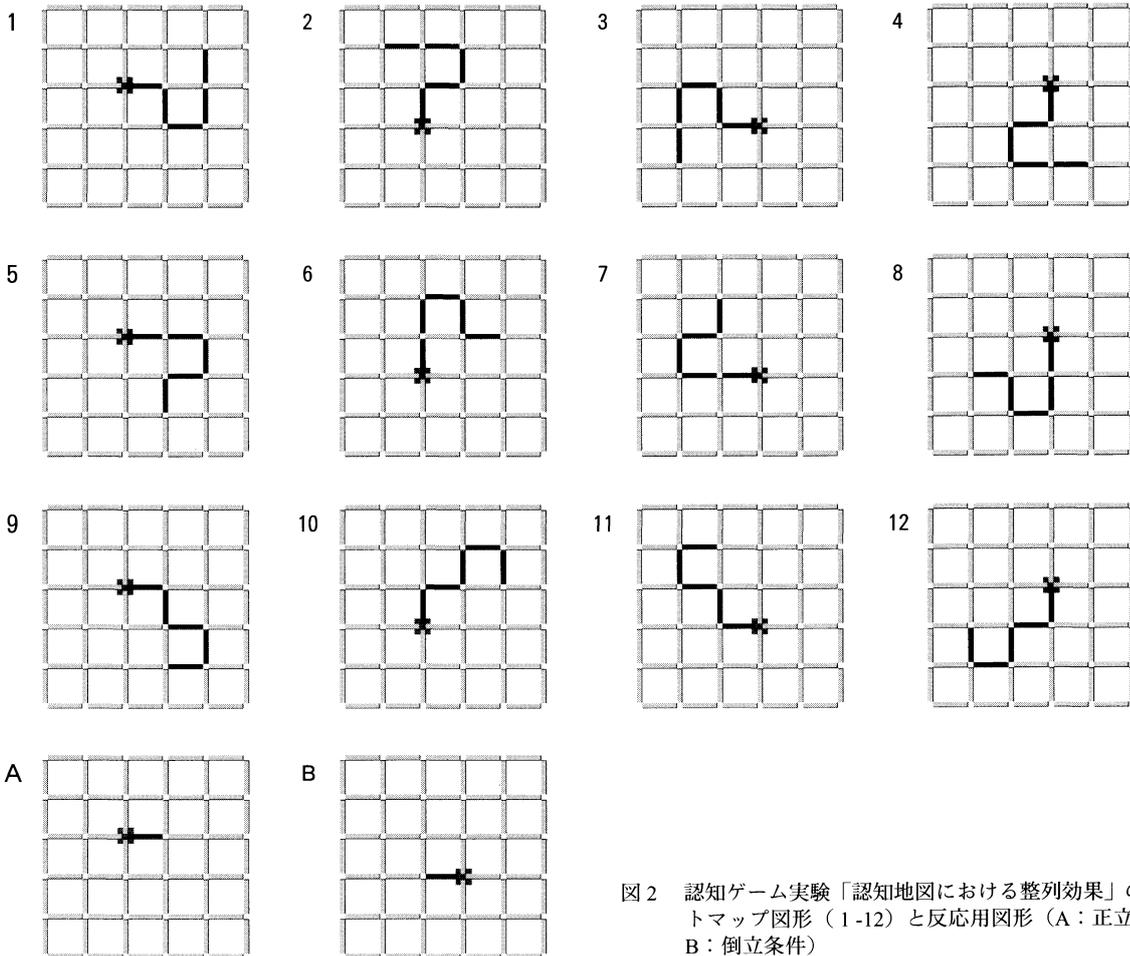
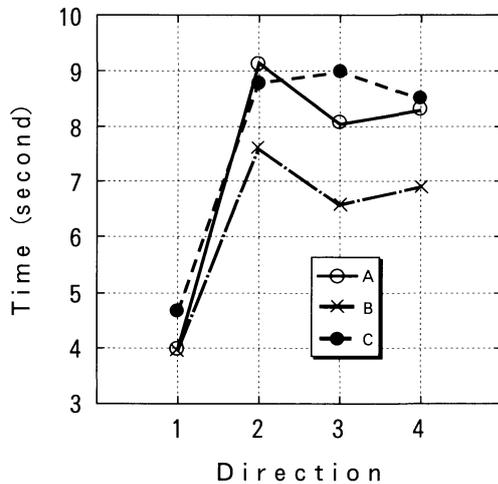


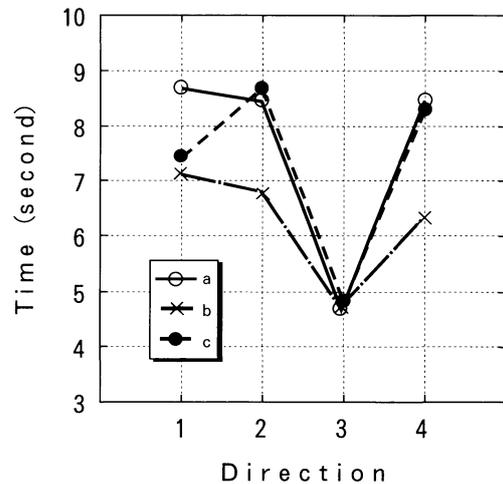
図2 認知ゲーム実験「認知地図における整列効果」のルートマップ図形（1-12）と反応用図形（A：正立条件、B：倒立条件）

表1 24刺激の平均反応時間とSD (ミリ秒)

(1) 正立図形条件					(2) 倒立図形条件				
No.	1	2	3	4	No.	13	14	15	16
mean	3966	9090	8134	8253	mean	8741	8534	4686	8383
SD	1680	8528	3416	4165	SD	4853	5982	1839	6225
No.	5	6	7	8	No.	17	18	19	20
mean	4011	7595	6626	6924	mean	7104	6752	4669	6299
SD	1425	6580	3346	3223	SD	4600	3452	2417	2891
No.	9	10	11	12	No.	21	22	23	24
mean	4660	8807	8986	8460	mean	7405	8696	4855	8345
SD	1917	8409	6840	4804	SD	3771	6198	2438	5116



(1) 正立図形条件



(2) 倒立図形条件

図3 正立と倒立の図形条件別に4方向と3図形を分けて示した平均反応時間(秒)4つの方向(1、2、3、4)は反時計回りにそれぞれ0、90、180、270度を意味する。図形(A、B、C)は正立図形条件の3図形を、図形(a、b、c)は倒立図形条件での3図形を示す。

トが表示されるのでその後を続けてください。表示地図と再現地図の方向が変わる場合や出発点の位置が変わる場合もありますが、ルートの順序と方向を正確に再現してください。できるだけ間違えないように、かつ、できるだけ早く完成してください。1試行のクリックの回数は常に4回です。まずはじめに練習してみましょう。」質問などに関しては必要に応じて口頭で対応した。

## 結果

はじめに刺激図形ごとの反応時間(ミリ秒)の36人分の平均とSDを求めた(表1)。刺激番号1から12までは学習段階で3種の図形(それぞれA、B、Cとする)を4つの方向(反時計回りに0、90、180、270度を数字で1、2、3、4と略す)に提示し、反応の開始位置の方向がそれぞれの図形の0度の位置のものである。ここではそれを仮に「正立図形条件」と呼ぶ。刺激番号13から24までは学習段階では1から12までと同じ方向で、反応図形の開始位置の方向がそれぞれの

図形の180度の位置のものであり、「倒立図形条件」と呼ぶ(それぞれa、b、cと記す)。つまり刺激図形1、5、9は0度提示方向での「整列条件」であり、刺激図形15、19、23は180度提示方向での「整列条件」となる。正立と倒立の図形条件別に4方向と3図形を分けて平均反応時間(秒)をグラフにしたものが、図3で(1)が正立図形条件、(2)が倒立図形条件である。

正立図形条件の結果である表1の(1)と図3の(1)の平均反応時間について3種の図形(A、B、C)と4提示方向(1、2、3、4)の独立2要因分散分析の結果では図形( $F(2,420)=3.06, p<0.01$ )と方向( $F(3,420)=15.82, p<0.01$ )の2主効果が有意で、その交互作用は有意とはならなかった。それぞれの主効果のライアン法による下位検定(5%水準)の結果では、3つ図形の間には差がなく、方向では方向1のみが他の3方向と有意な差を示した。つまり方向1の「整列条件」がどの図形でも他の方向よりも有意に反応時間が短いことが分かった。表1の(2)と図3の

表2 24刺激の平均スコアとSD (最大値は4)

(1) 正立図形条件					(2) 倒立図形条件				
No.	1	2	3	4	No.	13	14	15	16
mean	4.0	2.2	1.7	2.2	mean	1.7	2.4	3.3	2.1
SD	0.0	1.8	1.7	1.7	SD	1.8	1.7	1.4	1.9
No.	5	6	7	8	No.	17	18	19	20
mean	4.0	3.4	2.9	3.2	mean	3.1	3.2	3.9	3.1
SD	0.2	1.1	1.5	1.4	SD	1.5	1.4	0.7	1.5
No.	9	10	11	12	No.	21	22	23	24
mean	3.8	2.7	2.7	2.1	mean	2.8	2.6	3.8	3.0
SD	0.8	1.7	1.7	1.7	SD	1.7	1.7	0.7	1.6

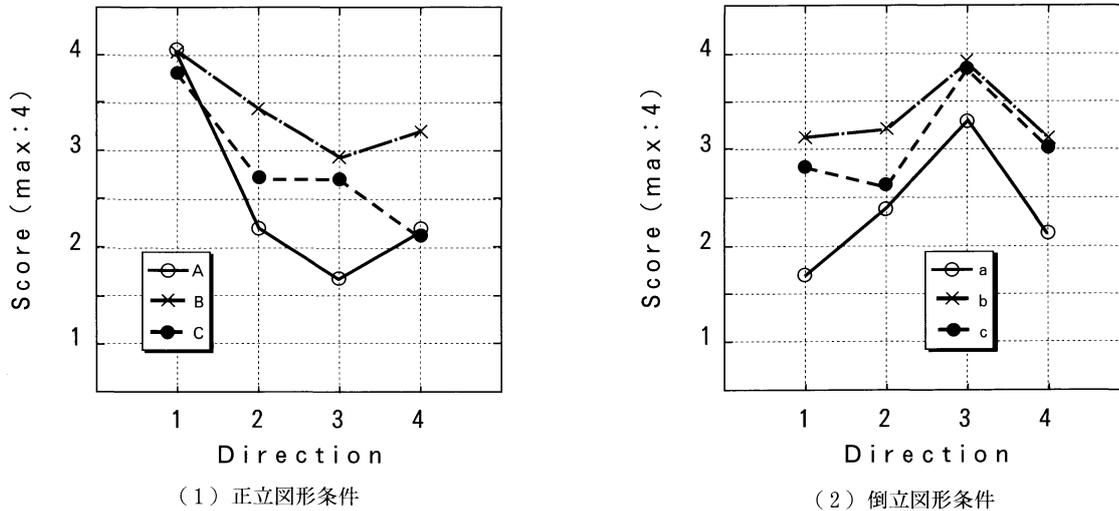


図4 正立と倒立の図形条件別に4方向と3図形を分けて示した平均スコア

(2) の倒立図形条件の平均反応時間について3種の図形 (a, b, c) と4提示方向 (1, 2, 3, 4) の独立2要因分散分析の結果では図形 ( $F(2,420)=3.98, p<0.01$ ) と方向 ( $F(3,420)=13.18, p<0.01$ ) の2主効果が有意で、その交互作用は有意とはならなかった。それぞれの主効果のライアン法による下位検定 (5%水準) の結果では、3つ図形の間で a-b 図形と b-c 図形間に差があり、方向では方向3のみが他の3方向と有意な差を示した。つまり倒立図形条件でも方向3の「整列条件」がどの図形でも他の方向よりも有意に反応時間が短いことが分かった。

次に刺激図形ごとのスコア (最大値は4) の36人分の平均とSDを求めた (表2)。正立図形条件の結果である表2の(1)と図3の(1)の平均スコアについて3種の図形 (A, B, C) と4提示方向 (1, 2, 3, 4) の独立2要因分散分析の結果では図形 ( $F(2,420)=13.22, p<0.01$ ) と方向 ( $F(3,420)=26.02, p<0.01$ ) の2主効果が有意で、その交互作用は有意とはならなかった。それぞれの主効果のライアン法による下位検定 (5%水準) の結果では、3つ図

形の間で A-B 図形と B-C 図形間に差があり、方向では方向1のみが他の3方向と有意な差を示した。つまり方向1の「整列条件」がどの図形でも他の方向よりも有意にスコアが高いことが分かった。表2の(2)と図3の(2)の倒立図形条件の平均スコアについて3種の図形 (a, b, c) と4提示方向 (1, 2, 3, 4) の独立2要因分散分析の結果では図形 ( $F(2,420)=14.63, p<0.01$ ) と方向 ( $F(3,420)=12.61, p<0.01$ ) の2主効果が有意で、その交互作用は有意とはならなかった。それぞれの主効果のライアン法による下位検定 (5%水準) の結果では、3つ図形の間で a-b 図形と a-c 図形間に差があり、方向では方向3のみが他の3方向と有意な差を示した。つまり倒立図形条件でも方向3の「整列条件」がどの図形でも他の方向よりも有意にスコアが高いことが分かった。

さらに24試行の間に試行の単純な練習効果と、基本ルートが3種であることに気づくことによって判断が容易になった可能性を平均反応時間で分析した。24試行を4つの段階に分けて6図形ずつ (ランダム順提示なので人によって図形番号は異なる) の平均反応時間

表 3 24試行を 4つの段階に区切った場合の平均反応時間 (単位はミリ秒)

order	1	2	3	4
1	9082	7277	6863	5732
2	8044	7297	7048	5900
3	8432	7127	7103	6105
4	8554	7842	7141	5464
5	8048	6377	6904	5549
6	7635	6727	6365	7365
mean	8299	7108	6904	6019
SD	459	463	261	638

の平均と SDを求めたのが、表 3である。時間の 4段階を被験者間 1 要因とする分散分析の結果、時間の効果が有意 ( $F(3,20)=19.56, p<0.01$ ) となり、ライアン法による下位検定 (5%水準)の結果から段階 2と 3の間以外には有意差が認められた。つまり大きく分けてみた時の反応時間は実験の後にあるほど早くなったといえる。

#### 4. ゲーム空間における認知地図

今回の実験では PC 画面上のグリッド状のルートパターンを用いた課題で、反応時間と正答率のデータから明確な「整列効果」が示された。ここでは PC 画面は 2次元パターンを表示し、内容も 2次元のマップとして理解されるものであった。ゲームの世界では多くの 3次元が描き出されることも多い。そこで、いわゆる「バーチャル・リアリティ」や「サイバースペース」あるいは「ゲーム空間」という仮想 3次元空間での「認知地図」の成立プロセスや利用のされ方の特徴などが、実在空間・日常空間での場合とどのように異なるのかという視点での「認知ゲーム実験」(山上, 2006, 2007, 2008)も可能であろう。ゲーム空間内での認知地図の利用についてはいわゆる「ゲーマー」たちがさまざまな経験をしてきていると思われるので、

その「暗黙的」な体験を明示化するような調査・実験の計画もこれから考えてみたい。

#### 引用文献

- 天ヶ瀬正博 1991 認知地図の方向性 人文論叢 (大阪  
市立大学大学院文学研究科) 20 95-115.
- 藤井秀夫・乾敏郎 1992 空間のイメージ 箱田 (編)  
認知科学のフロンティア II 第 1 章 サイエンス社  
Pp.9-38.
- 福田忠彦・渡辺利夫 1996 ヒューマンスケープ 日科  
技連
- ギフォード 2005 環境心理学 原理と実践 (上) 北  
大路書房
- グールド・ホワイト 1981 頭の中の地図 -メンタル・  
マップ- 朝倉書店
- 羽生和紀 2006 環境心理学 海保・楠見 (監修) 心理  
学総合事典 32章 朝倉書店 Pp. 638-645.
- 開一夫・松井孝雄 2001 空間認知と参照枠 イメージ  
と認知 乾・安西 (編) 岩波書店 Pp.61-90.
- 加藤孝義 1986 空間のエコロジー 新曜社
- Levin M., Jankovic I.N., & Palij, M 1982 Principles of  
spatial problem solving. *Journal of Experimental  
Psychology: General*, 111, 157-175.
- 森晃徳 1991 認知地図 箱田 (編) 認知科学のフロン  
ティア I 3 章 サイエンス社 Pp.73-101.
- 森敏昭・井上毅・松井孝雄 1995 グラフィック認知心  
理学 サイエンス社
- 森敏昭・中條和光 (編) 2005 認知心理学キーワード  
有斐閣
- 若林芳樹 1999 認知地図の空間分析 地人書房
- 山上 暁 2006 認知ゲーム実験 (1) 鏡映描写 甲南  
女子大学研究紀要 人間科学編 42 7-11.
- 山上 暁 2007 認知ゲーム実験 (2) 神経衰弱ゲーム  
甲南女子大学研究紀要 人間科学編 43 1-8.
- 山上 暁 2008 認知ゲーム実験 (3) ストループ効果  
甲南女子大学研究紀要 人間科学編 44 1-8.
- 横山勝樹 1997 環境の知覚・認知・行動 高橋・長澤・  
西出 (編) 環境と空間 3.1 Pp.89-118.

<pre> 1 //----- 2 //(HSP3) 迷路バー08-4(整列効果・再生法) 3 //      再生法のみ完成      (yamagamai akira 08214-) 4 //----- 5 #include "tmanage3.as" :// ミリ秒測定 gmes() 6 #pack "tmanage3.as" :// 実行ファイル作成 7 #include "imagebtn0.hsp" :// ボタンに画像を貼り付け 8 #pack "imagebtn0.hsp" :// 実行ファイル作成 9 //----- 10 buffer 1: picload "BLACK.bmp" :// 押された色 11 buffer 2: picload "GRAY.bmp" :// ボタンの色 12 xx0=680+40: yy0=600: screen 0,xx0,yy0,,200,40 :cls : randomize 13 objmode 2 :font "sytemfont",18 : objsize 200,36 14 pos 300,100 : BUTTON "乱数列" ,*rand 15 stop//----- </pre>		1. 準備
<pre> 16 *rand :// 1 から N1 まで N9 個ずつのランダム数列 (配列ss()に) 17 N1=24 :N9=1:i=0: ppp=0: z=18: font "sytemfont" t",z 18 repeat : p=rnd(N1)+1 :Q1(p)=Q1(p)+1 19     if Q1(p) &gt; N9 : continue 20     i=i+1: ppp=ppp+1 : SS(ppp)=p 21     if i= 7 :i= 8 22     if i= 14 :i=15 23     if i= 21 :i=22 24     pos 30,i*z :mes str(ppp)+"" :":pos 70,i*z :mes p : 25     if ppp &gt;= N1*N9 : :break 26 loop : pos 300,460 : BUTTON "書取り終了",*dimension 27 stop//----- </pre>		2. 乱数発生
<pre> 28 *dimension :cls: objsize 200,36:zz=99:// 配列変数の宣言 29 dim yn,zz : dim rt,zz : dim stim,zz : dim ans,zz 30 dim bn,zz : dim bn2,zz : sdim sss,zz : sdim p,zz 31 pic="" : ii=0: sss1="" : sss2="" : sss3="" : sss4="" : name="test" :// 初期値 32 sss5="" : repeat 24,1 : p(cnt)="" :loop 33 //----- </pre>		3. 配列宣言
<pre> 34 name="" :pos 30,60:mes "name" : input name :name="S"+str(name)//:被験者番号入力 35 pos 30,220 : BUTTON "実 験" ,*kkk1 36 pos 30,300 : BUTTON "練 習" ,*kkk2 37 pos 30,380 : BUTTON "検 査" ,*kkk3 38 stop //----- 39 *kkk1 :CD=1 :et ="e" : cls :wait 100 : goto *main: 40 *kkk2 :CD=9 :et ="p" : cls : goto *instruction : goto *main 41 *kkk3 :CD=1 :et ="t" : cls :wait 100 : goto *main: 42 stop//----- 43 *main:tt1=gettime(3)*60*60*24+gettime(4)*60*24+gettime(5)*60+gettime(6)://開始秒 44 goto*buttons </pre>		4. モード選択
<pre> 45 stop//----- 46 *instruction : color 200,200,120: boxf 00,00,xx0,yy0: color 00,00,120 47 pos 10,10: font "sytemfont",36: mes "&lt; ルート記憶実験 &gt;": mes 48 font "sytemfont",20: mes "ゲームの方法": mes 49 mes "ルートを覚える実験です。はじめに出た画面のルートを覚えて" 50 mes "あとの画面でそれを再現してください。":mes 51 mes "ルートを再現する時、出発点と最初のセグメントが表示されるので" 52 mes "その後を続けてください。":mes 53 mes "表示地図と再現地図の方向が変わる場合や出発点の位置が変わる場合" 54 mes "もありますが、ルートの順序と方向を正確に再現してください。":mes 55 mes "できるだけ間違えないように、かつ、できるだけ早く完成してください。" 56 mes "1試行のクリックの回数は常に4回です。" 57 mes "まずはじめに練習してみましょう":pos 480,400: BUTTON "進む" ,*kkk7 58 stop :*kkk7: cls : goto *main: return 59 stop//----- </pre>		5. 教示提示
<pre> 60 *buttons: // 刺激番号ボタン行列を表示する (smallest version) 61 ii=ii+1:pos 30,30:mes "&lt; 刺激番号 &gt;)// 試行回数カウンター// 下は配列変数クリア 62 repeat zz:stim(cnt)=0:ans(cnt)=0:bn(cnt)=0:bn2(cnt)=0:rt(cnt)=0:yn(cnt)=0:loop 63 x0=60 :y0=60:vn=4: hn=6: h=60: v=40: hh=h+7: vv=v+7 :// 原点とボタンサイズと間隔 64 repeat vn,1:i=cnt:repeat hn,1:j=cnt: bn=j+(hn)*(i-1) :// ボタン番号 1 から 65 objmode 2: color 000,000,000 :font "sytemfont",36:// ボタンの文字サイズ 66 objsize h,v :pos x0+(j-1)*hh,y0+(i-1)*vv+h:p1(bn)=0:button str(bn),*response2 67 loop :loop : // 下は24回目で「END」ボタンを出す命令 ://if ii &gt;= 24 : 68 font "sytemfont",18 69 if et!="p" :pos 020,yy0-80: objsize 160,40: button "全体終了",*finish 70 if et="p" :pos 020,yy0-80: objsize 160,40: button "練習終了",*kkk1 : stop 71 stop ://終ったボタンは消す? -----// テスト中 72 // repeat bn,1 :// bn2 は押されたことのフラグ変数 73 // color 000,000,000 :font "sytemfont",12: mes str(bn)+"" :str( bn2(bn)): 74 // if bn2(cnt)=1: clrobj cnt-1://loop:stop 75 *response2:bn2(stat+1)=1:nn=(stat+1):mes str(nn): goto *mm12://return </pre>		6. 地図格子

```

80  // nn はボタン番号+1(1-12), stno は学習ルート番号(1-24)
81  // start はスタート点指定 (1は左上、3は右下)
82  if CD=9 :goto *practice // 刺激の準備(配列stimにボタン番号を代入)
83  //pattern A
84  if nn >= 1 and nn <= 4 : ans= 42,15,48,16,10
85  if nn >=13 and nn <=16 : ans= 47,14,41,13,19
86  if stno= 1: stim=2,2,42,15,48,16,10
87  if stno= 2: stim=2,3,14,42,09,37,36
88  if stno= 3: stim=3,3,47,14,41,13,19
89  if stno= 4: stim=3,2,15,47,20,52,53
90  //pattern B
91  if nn >= 5 and nn <= 8 : ans= 42,43,16,48,21
92  if nn >=17 and nn <=20 : ans= 47,46,13,41,08
93  if stno= 5: stim=2,2,42,43,16,48,21
94  if stno= 6: stim=2,3,14,08,37,09,43
95  if stno= 7: stim=3,3,47,46,13,41,08
96  if stno= 8: stim=3,2,15,21,52,20,46
97  //pattern C
98  if nn >= 9 and nn <=12 : ans= 42,15,48,22,53
99  if nn >=21 and nn <=24 : ans= 47,14,41,07,36
100 if stno= 9: stim=2,2,42,15,48,22,53
101 if stno=10: stim=2,3,14,42,09,38,10
102 if stno=11: stim=3,3,47,14,41,07,36
103 if stno=12: stim=3,2,15,47,20,51,19
104 goto *mm122//-----
105 *practice:// pattern P (練習パターン4種)
106 if stno= 1 : stim=2,2,42,15,47,20,52 : ans= 42,15,47,20,52 : start=1
107 if stno= 2 : stim=2,2,42,09,38,10,16 : ans= 42,09,38,10,16 : start=1
108 if stno= 3 : stim=2,2,42,09,37,36,07 : ans= 47,20,52,53,22 : start=3
109 if stno= 4 : stim=2,2,42,15,47,46,19 : ans= 47,14,42,43,10 : start=3
110 *mm122://第1刺激(学習ルート)の提示
111 color 255,0,0:sx=stim(0):sy=stim(1):sh=30:aj=-10://色:開始位置:サイズ:調整値
112 boxf x0+hh*sx+aj,y0+vv*sy+aj,x0+sh+hh*sx+aj,y0+sh+vv*sy+aj://スタート点
113 //imagebtn 42, 1,00,00, h0, v0://ボタン色柄(黒)
114 repeat 5,2: if stim(cnt)>0: imagebtn stim(cnt),1,000,000,h0,v0 :wait 100: loop
115 //-----
116 *repro:// ここから 再生法 の場合の第2刺激(反応ルート)の表示
117 objsize 100,24 : pos 10,450
118 if et="p" :BUTTON gosub "再生",*grid0:// テスト試行
119 if et="p" :wait 300 :gosub *grid0 // 実験試行の提示時間設定
120 //objsize 100,24 : pos 110,450 : BUTTON gosub "recog",*recog
121 //objsize 100,24 : pos 210,450 : BUTTON gosub "next",*nexttrial
122 stop://-----
123 *grid0: cls : color 200,200,120: boxf 00,00,480,480:
124 :wait 100:pz=1: goto *grid// 再生法の場合の第2刺激の作成と提示
125 //-----
126 *recog:// ここから 再認法 の場合の第2刺激の作成 -----未完
127 pz=1: gosub *grid:// 第2刺激(再認ルート)の作成と提示
128 color 255,0,0:sx=stim(0):sy=stim(1):sh=30:aj=-10://色:開始位置:サイズ:調整値
129 boxf x0+hh*sx+aj,y0+vv*sy+aj,x0+sh+hh*sx+aj,y0+sh+vv*sy+aj://スタート点
130 //imagebtn 42, 1,00,00, h0, v0://ボタン色柄
131 repeat 5,2: imagebtn stim(cnt),1,000,000,h0,v0:loop
132 pz=1: objsize 100,24:pos 10,450 : BUTTON "yes",*ppp9
133 objsize 100,24:pos 110,450 : BUTTON "no",*ppp9
134 return://-----
135 *grid // ボタン行列のルート行列を表示する
136 cls : color 200,200,120: boxf 00,00,480,480:// 第2刺激(反応ボード)
137 h0=100: h1=6: v0=100: v1=6 // 画像の大きさと間隔
138 flag=0 :ccc=0 : cc=0: t1 = gmsec():// GAME開始ミリ秒測定
139 //-----
140 x0=52 :y0=52://描画原点
141 // 縦バー (no.0-29)-----
142 vn=5: hn=6: h=12: v=60: hh=h+v: vv=hh //バーサイズとグリッドサイズ
143 repeat vn,1: i=cnt: repeat hn,1: j=cnt:
144 bn=j+(hn)*(i-1)-1 // ボタン番号 0 から
145 pos 850,bn*20 :mes str(i)+"-"+str(j)+"-"+str(bn)
146 //objID, buffer, buf-x, buf-y, width, height
147 objsize h,v : pos x0+(j-1)*hh,y0+(i-1)*vv+h :p1(bn)=0: button "",*response
148 imagebtn bn, 2,00,00, h0, v0:// ボタン色柄(グレー)
149 loop :loop :
150 // 横バー (no.30-59)-----
151 vn=6: hn=5: h=60: v=12 :objsize h,v //バーサイズ
152 repeat vn,1: i=cnt: repeat hn,1: j=cnt:
153 bn=j+(hn)*(i-1)-1+30:// ボタン番号 30 から
154 pos 850,bn*20 :mes str(i)+"-"+str(j)+"-"+str(bn)

```

7. 記憶ルート  
作成と提示8. 練習用  
ルート9. 再生法の  
設定10. 反应用  
格子の表示

```

155 pos x0+(j-1)*hh+v.y0+(i-1)*vv :p1(bn)=0: button ""*response
156 imagebtn bn, 2,00,00,h0, v0:// ボタン色柄(グレー)
157 loop :loop : if pz=1 :gosub *point1 :else : return
158 stop://-----
159 *point1 ::// pz=1で 第2刺激(反応ボード)のスタート点(2種類)を作成
160 if start=1: color 255,0,0: sx=2:sy=2 :sh=30: aj=-10:// スタート位置1:サイズ調整値
161 if start=3: color 255,0,0: sx=3:sy=3 :sh=30: aj=-10:// スタート位置2:サイズ調整値
162 boxf x0+hh*sx+aj.y0+vv*sy+aj.x0+sh+hh*sx+aj.y0+sh+vv*sy+aj: //スタート点
163 if start=1: imagebtn 42, 1,00,00, h0, v0:// ボタン色柄(黒)
164 if start=3: imagebtn 47, 1,00,00, h0, v0:
165 return: stop://-----
166 *response:cc=stat:// ボタンが押された場合の処理(cc = ボタン番号 0 から)
167 if p1(cc)=0:imagebtn cc,1,000,000,h0,v0:p1(cc)=1 :goto *ppp9:// 表(black)
168 if p1(cc)=1:imagebtn cc,2,000,000,h0,v0:p1(cc)=0 :goto *ppp9:// 裏(gray)
169 stop://-----
170 *ppp9 :ccc=ccc+1 //:cc=stat // クリック数カウンター
171 color 200,200,120 : boxf 0,0,480,20 // 結果表示クリア
172 t9 = gmsec(): rt(ccc)=t9 - t1 // 開始からの時間(ミリ秒)
173 bn(ccc)= cc: // ボタン番号記録
174 if bn(ccc) = ans(ccc) : yn(ccc) = 1 // 正答ならyn=1
175 font "sytf pos 10,0 :
176 if et!="e" : mes "ccc="+ccc+" bno.="+bn(ccc)+" rt=" +rt(ccc)+" : "+yn(ccc)
177 font "sytf pos 480,0+30*ccc :
178 mes str(nn)+" "+str(ccc)+" "+str(bn(ccc))+" "+str(rt(ccc)):// 1試行分記録
179 font "sytemfont",16,0 : // et="e"://テスト用
180 objsize 80,24 : //pos 490,400 : BUTTON gosub "試行記録" *record
181 //pos 590,400 : BUTTON gosub "チェック" *all2
182 pos 490,450 : BUTTON gosub "スコア" *nexttrial
183 //pos 590,000 : BUTTON gosub "終了" *finish
184 stop:return
185 *all2: all=1:color 255,000,000: goto *record2 //-----テスト中
186 *nexttrial : goto *record:
187 // *nexttrial : gosub *record: pz=0 : cls : goto *buttons
188 *finish: if et="e" : all=1: gosub *record2 : stop : else :end
189 stop://-----
190 *record: pos 460,10 :// 配列変数(1次元)をCSV形式で保存する
191 font "sytemfont",16 : color 255,000,000 :// 赤で表示
192 n= n+1: bb=10 : bb=18: CR= "r" : // CRLF="r"
193 tt7= gettime(3)*60*60*24+gettime(4)*60*24+gettime(5)*60+gettime(6):// 終了秒測定
194 tt8= str(tt7-tt1): // 全所要時間秒測定 // tt9 は記録ファイルのヘッダー
195 tt9= name + " "+str(gettime(0))+" "+str(gettime(1))+" "+str(gettime(3))
196 tt9= tt9 + " "+str(gettime(4))+" "+str(gettime(5))+" "+str(gettime(6))
197 tt9= tt9 + "(" +tt8+ " sec." + "CD=" + CD
198 nnn="no." : ppp = "" : kkk=0
199 iii= "番号" + " "+str(ccc)+" "+str(bn(ccc))+" "+str(rt(ccc))+" "+str(yn(ccc))+" "+str(yn(ccc))
200 iii=iii+" 正誤" + " "+str(ccc)+" "+str(bn(ccc))+" "+str(rt(ccc))+" "+str(yn(ccc))+" "+str(yn(ccc))
201 sss2 = tt9 + CR +CR+ iii+ CR: // ssss は全試行分の記録ストリング
202 repeat ccc,1 :// 全試行の1行分のデータを累積
203 nnn = nnn + str(cnt) + " " :// クリック番号系列
204 sss = sss + bn(cnt) + " " :// ボタン番号系列
205 aaa = aaa + ans(cnt) + " " :// 正答番号系列
206 if bn(cnt) = ans(cnt) : yy=yy+1 : yn(cnt) = 1 :// 正答なら1 (yy はスコア)
207 bbb = bbb + yn(cnt) + " " :// 正誤記録系列
208 rrt = rrt + rt(cnt) + " " :// 反応時間系列
209 if cnt != 1 : kkk = rt(cnt) - rt(cnt-1)
210 // ppp は1試行分の記録ストリング
211 ppp=ppp+str(nnn)+" "+str(cnt)+" "+bn(cnt)+" "+ans(cnt)+" "+
212 ppp=ppp+yn(cnt)+" "+rt(cnt)+" "+kkk+CR
213 p(nn)=p(nn)+str(nnn)+" "+str(cnt)+" "+bn(cnt)+" "+ans(cnt)+" "+yn(cnt)+" "+rt(cnt)+" "+kkk+CR
214 sss3= sss3 + ppp:// sss3 に記録を累積
215 //notesel sss3:noteadd ppp,0,1:// 1 は上書きモード
216 pos 480,10+cnt*16:if et!="e":mes ppp:// クリックごとの表示
217 ppp = "" :loop // ppp をクリア
218 //-----
219 sss4="刺激番号" + " "+str(ccc)+" "+str(bn(ccc))+" "+str(rt(ccc))+" "+str(yn(ccc))+" "+str(yn(ccc))
220 sss1= sss1+ str(n)+" "+str(n)+" "+str(yy)+" "+str(rt(4)) + CR
221 sss3= sss3 +CR: if et!="e" : mes "-----" :mes sss3:
222 font "sytemfont",16 : color 000,255,000: if et="e" : mes sss1 :// 緑で表示
223 color 000,000,255 :if et="e" : mes p(nn) :// 青で表示
224 font "sytemfont",16 : color 255,000,000: // 赤で表示
225 color 0,0,0:pos 500,180: // 黒で表示
226 font "sytemfont",36 :mes:mes "スコア : "+str(yy) :yy=0: //wait 200
227 pos 590,500 : font "sytemfont",16: BUTTON goto "次試行" *next0 : stop
228 *next0 : pz=0 : cls : wait 100: goto *buttons
229 //return ://if et="e" and all=1 :goto *record2 // :else :goto *buttons
230 return://-----
231 *record2: font "sytemfont",16 : color 255,000,000 :pos 490,0 :// 赤で表示
232 repeat 24,1: sss5=sss5+p(cnt): loop: sss5 = "番号順"+CR+sss5+CR
233 if all=1 :sss3=sss2+sss3+sss4+sss1+sss5:mes "files created" :mes sss3://stop
234 //font "sytemfont",16 : color 000,255,000 :mes sss1 :sss1=""
235 name="S"+name :font "sytemfont",16 : color 255,000,000
236 if et="e" : notesel sss3: notesave name+".csv":// 実験時のみ記録ファイル作成
237 return:stop://-----

```

11. 反応出発点の表示

12. 反応入力処理

13. データファイルの作成