Vol.2010-MPS-77 No.27
2010/3/5

時空間情報と動作を組み合わせた認証方法

角田雅照[†1] 伏田享平[†1] 三井康平[†1]
亀井靖高[†1] 中村匡秀[†2]
後藤慶多[†1] 松本健一[†1]

本稿では，時空間情報（位置，移動時間，移動距離）と動作と組み合わせた認証方法を提案する．ユーザは時空間情報で定義された特定の認証点において，特定の動作を行うことを繰り返すことにより認証に成功する．さらに本稿では，ユーザの認証行為の部分的な誤りを許容する，部分一致認証を提案するとともに，時空間情報と動作の組を時空間文字と定義し，時空間文字に基づいて認証方法の安全性の評価を行う．実験により提案方法の安全性を評価した結果，本人拒否率は 0.003%，他人受入率は 0.019%となった．

An Authentication Method with the Combination of Spatiotemporal Information and Actions

Masateru Tsunoda ,[†1] Kyohei Fushida ,[†1]
Kohei Mitsui ,[†1] Yasutaka Kamei ,[†1]
Masahide Nakamura ,[†2] Keita Goto [†1]
and Ken-ichi Matsumoto[†1]

We propose a new authentication method combining actions and spatiotemporal information such as location, elapsed time, and travel distance of a user. To be authenticated, a user performs certain actions at certain points defined with spatiotemporal information. Additionally, we propose partial matching authentication which allows partial mistake of user's authentication procedure, and define combination of spatiotemporal information and action as spatiotemporal character. With spatiotemporal character, we evaluate security of our proposed authentication method. We experimented to confirm the security of the proposed method, and the results showed that result false rejection rate is 0.003% and false acceptance rate is 0.0019%.

1. はじめに

近年，安全性が高い認証方法として，IC カードを用いた認証方法と生体情報を用いた認証方法が普及しつつある．ただし，厳格な認証が要求される場面（機密データ保管室，軍事施設，原子力発電所などの重要施設に入場する際の認証）では，これらの認証方法だけでは十分であるとはいえない．IC カードを用いた認証方法では，IC カードが盗難される危険性がある．その場合，IC カードを利用不能にすればよいが，ユーザが盗難に気づく前に認証に利用されてしまう危険性がある[2]．生体情報を用いた認証方法の場合，盗難される可能性はないが，人工物により不正に認証される問題が指摘されており[3),5)]，実際に指紋認証をすり抜ける事件も発生している[7]．

これらの問題点を補う最も簡単な方法は，2 つの認証方法を併用する，二要素認証を用いることである．二要素認証では,「何を知っているか」に基づく認証方法と「何を持っているか」に基づく認証方法を組み合わせるのが通常である[6]．「何を知っているか」に基づく認証方法として，最も一般的な認証方法はパスワードに基づく認証方法であるが，他人に入力を盗み見られること（のぞき見攻撃）によりパスワードを知られてしまう危険性がある．のぞき見攻撃に強い認証方法として，ユーザの動作（携帯端末を動かした軌跡や歩き方など）[1)4)] を用いた認証方法があげられるが，生体情報を用いた認証方法ほどの安全性は実現できていない．

そこで本稿では，**時空間情報**（ユーザの位置，移動時間，移動距離など）に着目し，動作と組み合わせることにより安全性を高めた認証方法を提案する．提案方法では，ユーザの時空間情報が特定の**認証点**と一致するときに，ユーザが特定の動作を行うことを繰り返すにより認証される．**図 1** に地点と携帯ボタンデバイスを組み合わせた認証例を示す．認証方法を知っているユーザは，駅を出発して地点 A, 地点 B，地点 C，地点 D で携帯するボタンを押下し，すべて正しければデータセンターに入ることができる．本稿では．時空間情報を認証に用いるために，ユーザが認証行為を部分的に誤ることを考慮した認証方式（部分一致認証）を提案するとともに，時空間情報と動作に基づく認証方法の安全性の評価方法を明らかにする．

†1 奈良先端科学技術大学院大学 情報科学研究科
Graduate School of Information Science, Nara Institute of Science and Technology
†2 神戸大学 大学院工学研究科
Graduate School of Engineering, Kobe University

Vol.2010-MPS-77 No.27
2010/3/5

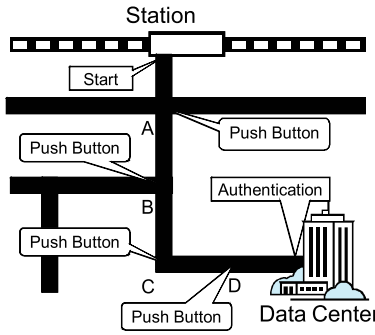

図 1 地点情報とボタン押下動作を組み合わせた認証例
Fig. 1 Example of Authentication with the Combination of the Location Information and Pushing Button Action.

以降，2 章において提案方法の詳細について述べ，3 章で安全性の評価方法について説明する．4 章で安全性を評価した実験について述べ，5 章でまとめと今後の課題について述べる．

2. 時空間情報と動作を用いた認証方法

2.1 時空間情報の定義

時空間情報とは一般に,「いつ」（時間）と「どこ」（空間）がひも付けられた情報（の系列）を指す．本稿では，ユーザ U の認証開始時からの**位置**，**距離**，**時間**の 3 種類の情報およびそれらの組を時空間情報と定義し，認証に用いることにする．

位置 位置 pos は，ある時点においてユーザ U がいる場所であり，緯度，経度，高度で定義される．

地点 認証に用いるためにシステムであらかじめ決められた場所を地点 p_i と呼ぶ．パスワード入力に例えると，地点はシステムパスワードであり，位置はユーザが実際に入力したパスワードにあたる．位置と同様，緯度，経度，高度で定義される．$pos = p_i$ のとき，U は p_i を**通過した**という． U が認証開始時に通過した地点を p_0 と書く．

時空間情報 U が p_0 を通過後，ある時点 t において場所 pos_t に到達したと仮定する．また，その時の移動距離が $dist_t$, 移動時間が $etime_t$ であった時，U の t における時空間情報 $spctmp_t$ を次のように定義する．

$$spctmp_t := (pos_t, etime_t, dist_t) \quad (1)$$

2.2 認証点の定義

次に，時空間情報 $spctmp$ を用いて認証を行う際の認証点という概念を定義する．

地点認証点 p_i を任意の地点とし，この地点で定義される認証点 LP を地点認証点と呼ぶ．例えば，図 1 の例において，各地点（A〜D）は LP である．

時間認証点 $etime_i$ を任意の移動時間とし，この移動時間で定義される認証点 TP を時間認証点と呼ぶ．時間認証点は，U の現在の移動時間が $etime_i$ の時点と解釈される．例えば，TP が 15 秒とは，認証開始後 15 秒の時点を表す．

距離認証点 $dist_i$ を任意の移動距離とし，この移動距離で定義される認証点 DP を距離認証点と呼ぶ．距離認証点は，U の現在の移動距離が $dist_i$ の時点と解釈される．例えば，DP が 100m とは，認証開始後 100m の時点を表す．

LP，TP または DP を特に区別する必要が無い文脈ではそれらを総じて**認証点**と呼び，P と書く．

2.3 認証動作の定義

認証動作とは，認証点 P においてユーザ U が認証のために行うべき動作である．任意の認証動作 Act に対して，Act の取りうる値の集合を range(Act)，システムであらかじめ規定された動作を**認証動作値** $a_s(a_s \in \text{range(Act)})$，ユーザ U が実際に行った動作を**入力動作** a_e と呼ぶ．図 1 の例では「ボタンを押下する」という動作が認証動作 Act に相当する．提案手法では，認証動作として複数の状態をとることができる任意の動作を採用可能である．

2.4 認証文字列，入力文字列の定義

認証点 P と認証動作 Act を用いて，認証文字列，入力文字列を定義する．

認証文字列 認証文字列は，あらかじめ決められた認証点と認証動作値のペアの系列として定義される．すなわち，

$$\text{Auth\_Str} = \langle \mathrm{P}_1, a_1\rangle\langle \mathrm{P}_2, a_2\rangle \cdots \langle \mathrm{P}_i, a_{si}\rangle \cdots \langle \mathrm{P}_k, a_{sk}\rangle \quad (2)$$

ここで，P_i は認証点，a_{si} は認証動作値である．$\langle \mathrm{P}_i, a_{si}\rangle$ を**時空間文字**と呼ぶ．この定義に基づくと，図 1 の例は以下のように定式化できる．

Auth_Str1 = ⟨A,“ボタンを押す”⟩⟨B,“ボタンを押す”⟩⟨C,“ボタンを押す”⟩⟨D,“ボタンを押す”⟩

入力文字列 ユーザ U が p_0 を通過後，k 回の入力動作 $a_{e1}, a_{e2}, \cdots, a_{ei}, \cdots, a_{ek}$ を行ったとする．また，a_{ei} を実行した時の，U の時空間情報を $spctmp_i$ とする．このとき，$spctmp_i$ と a_{ei} の組の系列を入力文字列という．すなわち，

$$\text{Input} = \langle spctmp_1, a_{e1}\rangle\langle spctmp_2, a_{e2}\rangle \cdots \langle spctmp_i, a_{ei}\rangle \cdots \langle spctmp_k, a_{ek}\rangle \quad (3)$$

時空間情報 $spctmp$ と認証点 P が与えられた時，それらの定義から，時空間情報が認証点上か認証点外かを判定することができる．「$spctmp$ が認証点 P 上にある」という関係を $spctmp\ AT$ P と書く．また，「$spctmp$ が認証点 P 外にある」という関係を $spctmp\ OUT$ P と書く．

2.5 部分一致認証

部分一致認証は，入力文字列に誤りが含まれていることを考慮した方法である．認証文字列 Auth_Str とユーザ U の入力文字列 Input との不一致の時空間文字が tn 個以下ならば，認証に成功していると判定する．tn を**不一致許容数**と呼ぶ．Auth_Str に対し，U の Input が以下の条件を満たす時，かつそのときに限り，U は認証を成功する．

$$(\exists i(1 \leq i \leq in)\ \exists j(1 \leq j \leq n); spctmp_i\ AT\ \mathrm{P}_j \wedge a_{ei} = a_{sj}) \\ \wedge\ mn \geq n - tn\ \wedge\ in \leq n \qquad (4)$$

ここで n は Auth_Str の時空間文字数，in は Input の時空間文字数，mn は $spctmp_i AT \mathrm{P}_j \wedge a_{ei} = a_{sj}$ が成り立っている時空間文字の数を表す．in は n 以下である必要がある（大量の時空間文字が入力され，時空間文字が一致する可能性が高まることを防ぐため）．より直感的には，時空間文字 $\langle \mathrm{P}_i, a_{si} \rangle$ がシステムパスワードの 1 文字に相当し，U は時空間情報 $spctmp_i$ と入力動作 a_{ei} の組み合わせで表された時空間文字 $\langle spctmp_i, a_{ei} \rangle$ を入力していくと考えればよい．パスワード入力に例えると，システムパスワードが n 文字の場合，ユーザが入力した文字がシステムパスワードと $n - tn$ 個以上一致していれば（ユーザのタイプミスが tn 文字以下ならば），認証成功と判定する．

なお，入力文字列 Input に未入力文字が含まれる可能性（時空間文字の入力をスキップしてしまうこと）を考慮し，入力文字列 Input のある文字が，認証文字列 Auth_Str のいずれかの文字に一致している場合，その時空間文字が一致していると判定している．パスワードに例えると，システムパスワードが “ABCD”， 不一致許容数が 1 の場合，ユーザの入力文字が “ABCE”，“ABC”，“BCDE”，“BCD” などの場合，認証に成功する

2.6 認証点マージンの定義

ユーザ U の入力動作が認証点 P から多少ずれることを考慮して，認証点マージンを定義する．認証点を P，時空間情報を $spctmp$，$Margin_p$ を P の前方のマージン，$Margin_a$ を認証点の後方のマージンとするとき，$\mathrm{P} - Margin_p \leq spctmp < \mathrm{P} + Margin_a$（地点認証点 LP の場合，認証点から半径 $Margin$ 以内に $spctmp$ が含まれる）ならば，$spctmp\ AT$ P と判定する．特に認証点マージンを区別する必要がある場合，時間認証点マージンを $tMargin_p$，$tMargin_a$，距離認証点マージンを $dMargin_p$，$dMargin_a$，地点認証点マージンを $pMargin$ と表す．例えば，時間認証点 TP を認証開始 30 秒後とし，$tMargin_p$，$tMargin_a$ を 2 秒とした場合，28 秒以上 32 秒未満の範囲が TP とみなされ，その範囲において行った動作は，TP において行った動作とみなされる．

3. 安全性の評価方法

本章では，提案方法の安全性の評価方法について説明する．提案方法の安全性は，認証文字列 Auth_Str と入力文字列 Input が偶然一致する確率に基づく．この確率の分母は Auth_Str のパターン数，分子は Input のパターン数となる．まず，安全性の評価のための用語の定義と，認証点の候補数について説明し，その後 Auth_Str のパターン数と Input のパターン数に基づき，Auth_Str と Input が偶然一致する確率を導出する．

3.1 用語の定義

要素入力失敗率 認証文字列 Auth_Str を知っているユーザ U が，時空間文字 $\langle spctmp_i, a_{ei} \rangle$ の入力を失敗する確率．パスワード入力に例えると，システムパスワードを知っているユーザが，1 文字入力する際にタッチミスをする確率を指す．

文字列入力失敗率 認証文字列 Auth_Str を知っているユーザ U が，入力文字列 Input を正しく入力できなかったために，認証に失敗する確率．パスワード入力に例えると，システムパスワードを知っているユーザが，入力時にタイプミスがあったために，認証に失敗する確率を指す．要素入力失敗率を pa，認証文字列 Auth_Str に含まれる時空間文字の個数を m，不一致許容数（2.5 参照）を tn とすると，文字列入力失敗率 pf は，時空間文字 m 個のうち $tn+1$ 文字以上，時空間文字の入力に失敗する確率であり，以下のように表される（pf は二項分布に従う）．

$$pf = \sum_{i=tn+1}^{m} {}_m\mathrm{C}_i\ pa^i (1-pa)^{m-i} \qquad (5)$$

試行可能回数 入力文字列 Input の入力を試行することができる回数．試行可能回数以内に入力文字列 Input の入力を成功させない場合，認証が拒否される．パスワード入力に例えると，入力を何回まで試みることができるかを表す．

本人拒否率（FRR） 認証文字列 Auth_Str を知っているユーザ U が，試行可能回数以内に入力文字列 Input の入力を成功させることができず，認証を拒否される確率を本人拒否率（False Rejection Rate; FRR）とする．文字列入力失敗率を pf，試行可能回数を r とするとき，FRR は以下の式により計算される（pf の具体的な値は 4 章の実験

Vol.2010-MPS-77 No.27
2010/3/5

表 1 例で用いる認証文字列の設定
Table 1 Settings of Authentication String used in examples.

パラメータ	値
認証点 P の候補	A から Z; 26 種類
認証動作 Act の候補	0 から 9; 10 種類
時空間文字の候補	⟨[A-Z][0-9]⟩
認証文字列 Auth_Str	⟨A0⟩⟨B0⟩⟨C0⟩⟨D0⟩

において確かめる）.

$$\mathrm{FRR} = pf^r \tag{6}$$

入力文字列一致率 攻撃者の i 回目の認証試行における入力文字列 Input が，偶然認証文字列 Auth_Str に一致し，攻撃者が認証に成功する確率.

他人受入率（FAR） 攻撃者が入力文字列 Input の入力を r 回繰り返して偶然認証に成功する確率を，他人受入率（False Acceptance Rate; FAR）とする．入力文字列一致率を pt_i とすると，FAR は以下のように表される.

$$\mathrm{FAR} = 1 - \prod_{i=1}^{r}(1 - pt_i) \tag{7}$$

3.2 認証点の候補数

認証点の候補数は，認証時に移動する距離や時間によって決まる．例えば時間認証点の場合，認証にかける時間を $totalTime$ とし，各認証点を $tMargin_p + tMargin_a$ 以上離すとすると，認証点の候補数 cn は以下のようになる.

$$cn = \left[\frac{totalTime}{tMargin_p + tMargin_a} + 1\right] \tag{8}$$

例えば，$totalTime$ を 100 秒，$tMargin_p$，$tMargin_a$ を 1 秒とすると，cn は $[100 \div (1 + 1) + 1] = 51$ となる.

3.3 入力文字列一致率

入力文字列一致率 pt_i の導出方法について説明する．確率導出の理解を容易にするため，まず具体的な例を用いて説明し，その後一般化した数式を示す．例では**表 1** のように認証文字列 Auth_Str を設定し，時空間文字の候補を正規表現を用いて表す.

Auth_Str と一致しない時空間文字は，以下の 2 つに分類することができる.

- 「認証点が不一致，動作が一致または不一致」の場合．時空間文字は正しい認証点 [A-D] を除いた ⟨[E-Z][0-9]⟩ となる.
- 「認証点が一致，動作が不一致」の場合．時空間文字は ⟨[A-D][1-9]⟩ となる.

また，部分一致認証（不一致許容数が $tn > 0$）は，以下のように場合分けされる.

- 入力文字列 Input と認証文字列 Auth_Str が一致しない文字が tn ある（未入力文字が存在しない）場合（不一致許容数が 2 の場合，Input が ⟨A0⟩⟨B0⟩⟨[E-Z][0-9]⟩⟨[E-Z][0-9]⟩ など）．この場合を**ケース 1** と呼ぶこととする.
- 入力文字列 Input と認証文字列 Auth_Str が一致しない文字と，未入力の文字の合計が tn ある（未入力文字が存在する）場合（不一致許容数が 2 で未入力文字が 1 文字の場

表 2 未入力文字が存在しない場合の，認証に成功する入力文字列のパターン数
Table 2 Number of Patterns of Successful Input String with No Missing Character.

不一致許容数	⟨[E-Z][0-9]⟩の文字数	⟨[A-D][1-9]⟩の文字数	⟨[A-D][0-9]⟩のパターン数	⟨[A-D][1-9]⟩のパターン数	⟨[E-Z][0-9]⟩のパターン数
0	0	0	${}_4\mathrm{C}_4$	${}_4\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_0 \times 10^0$
1	1	0	${}_4\mathrm{C}_3$	${}_3\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_1 \times 10^1$
	0	1	${}_4\mathrm{C}_4$	${}_4\mathrm{C}_1 \times 9^1$	${}_{22}\mathrm{C}_0 \times 10^0$
2	2	0	${}_4\mathrm{C}_2$	${}_2\mathrm{C}0 \times 9^0$	${}_{22}\mathrm{C}_2 \times 10^2$
	1	1	${}_4\mathrm{C}_3$	${}_3\mathrm{C}1 \times 9^1$	${}_{22}\mathrm{C}_1 \times 10^1$
	0	2	${}_4\mathrm{C}_4$	${}_4\mathrm{C}_2 \times 9^2$	${}_{22}\mathrm{C}_0 \times 10^0$
3	3	0	${}_4\mathrm{C}_1$	${}_1\mathrm{C}0 \times 9^0$	${}_{22}\mathrm{C}_3 \times 10^3$
	2	1	${}_4\mathrm{C}_2$	${}_2\mathrm{C}1 \times 9^1$	${}_{22}\mathrm{C}_2 \times 10^2$
	1	2	${}_4\mathrm{C}_3$	${}_3\mathrm{C}_2 \times 9^2$	${}_{22}\mathrm{C}_1 \times 10^1$
	0	3	${}_4\mathrm{C}_4$	${}_4\mathrm{C}_3 \times 9^3$	${}_{22}\mathrm{C}_0 \times 10^0$

表 3 未入力文字が存在する場合の，認証に成功する入力文字列のパターン数
Table 3 Number of Patterns of Successful Input String with Missing Characters.

不一致許容数	未入力文字数	⟨[E-Z][0-9]⟩の文字数	⟨[A-D][1-9]⟩の文字数	⟨[A-D][0-9]⟩のパターン数	⟨[A-D][1-9]⟩のパターン数	⟨[E-Z][0-9]⟩のパターン数
1	1	0	0	${}_4\mathrm{C}_3$	${}_3\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_0 \times 10^0$
2	1	1	0	${}_4\mathrm{C}_2$	${}_3\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_1 \times 10^1$
		0	1	${}_4\mathrm{C}_3$	${}_3\mathrm{C}_1 \times 9^1$	${}_{22}\mathrm{C}_0 \times 10^0$
	2	0	0	${}_4\mathrm{C}_2$	${}_2\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_0 \times 10^0$
3	1	2	0	${}_4\mathrm{C}_1$	${}_1\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_2 \times 10^2$
		1	1	${}_4\mathrm{C}_2$	${}_2\mathrm{C}_1 \times 9^1$	${}_{22}\mathrm{C}_1 \times 10^1$
		0	2	${}_4\mathrm{C}_3$	${}_3\mathrm{C}_2 \times 9^2$	${}_{22}\mathrm{C}_0 \times 10^0$
	2	1	0	${}_4\mathrm{C}_1$	${}_1\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_1 \times 10^1$
		0	1	${}_4\mathrm{C}_2$	${}_2\mathrm{C}_1 \times 9^1$	${}_{22}\mathrm{C}_0 \times 10^0$
	3	0	0	${}_4\mathrm{C}_1$	${}_1\mathrm{C}_0 \times 9^0$	${}_{22}\mathrm{C}_0 \times 10^0$

合，Input が ⟨A0⟩ ⟨B0⟩ ⟨[E-Z][0-9]⟩ など）．この場合を**ケース 2** と呼ぶこととする．

a　認証文字列 Auth_Str のパターン数

(a-1) 例に基づくケース 1 のパターン数　時間認証点 TP，距離認証点 DP，及び地点認証点 LP でルートが 1 つだけの場合，Auth_Str において認証点 P の重複はなく，また，P の出現順序も一定のため，Auth_Str に含まれる P のパターン数は，26 文字から 4 文字を取り出す組み合わせの総数 ${}_{26}\mathrm{C}_4$ となる．認証動作値 a_s は P ごとに設定され，別の P において同じ a_s を設定可能である．よって，Auth_Str のパターン数は ${}_{26}\mathrm{C}_4 \times 10^4$ となる．

(a-2) 例に基づくケース 2 のパターン数　不一致許容数が 1，未入力文字が 1 文字の場合，Auth_Str のパターン数は，ケース 1 のパターン数 ${}_{26}\mathrm{C}_4 \times 10^4$ に時空間文字が 3 文字の場合の数 ${}_{26}\mathrm{C}_3 \times 10^3$ を加えたものとなる．

(a-3) (a-1)，(a-2) の一般化　未入力文字は最大で tn となるため，不一致許容数が tn のとき，Auth_Str のパターン数は $\sum_{j=0}^{tn} {}_{cn}\mathrm{C}_{n-j}\, en^{n-j}$ となる．ここで，n は Auth_Str に含まれる時空間文字の数，cn は認証点の候補数，en は認証動作のパターン数である．

b　認証に成功する入力文字列 Input のパターン数

(b-1) 例に基づくケース 1 のパターン数　不一致許容数を 0 から 3 に変化させたときの，認証に成功する入力文字列 Input のパターン数を**表 2** に示す．不一致許容数ごとに，⟨[E-Z][0-9]⟩（認証点が不一致，動作が一致または不一致）と ⟨[A-D][1-9]⟩（認証点が一致，動作が不一致）の文字数の違いによって場合分けができる．

ここで，表 2 の ⟨[A-D][0-9]⟩ のパターン数とは，認証点が一致している（動作の一致，不一致は考慮しない）時空間文字のパターン数である．例えば不一致許容数が 2，⟨[E-Z][0-9]⟩ の文字数が 1，⟨[A-D][1-9]⟩ の文字数が 1 のとき（表 2 の 5 行目），Auth_Str ⟨A0⟩ ⟨B0⟩ ⟨C0⟩ ⟨D0⟩ の 4 文字から 3 文字を選ぶ場合の数であり，${}_4\mathrm{C}_3$ となる．

表 2 の ⟨[A-D][1-9]⟩ のパターン数とは，認証点が一致，動作が不一致となる時空間文字のパターン数である．前述の例と同様の条件の場合，Auth_Str ⟨A0⟩ ⟨B0⟩ ⟨C0⟩ ⟨D0⟩ の 4 文字から 3 文字を選んだ後，その 3 文字から ⟨[A-D][1-9]⟩ となる 1 文字を選ぶ場合の数 ${}_3\mathrm{C}_1$ に，[1-9] の組み合わせ 9^1 を乗じた ${}_3\mathrm{C}_1 \times 9^1$ となる．

(b-2) 例に基づくケース 2 のパターン数　不一致許容数を 0 から 3 に変化させ，かつ未入力文字数も変化させた場合の，認証に成功する入力文字列 Input のパターン数，および Auth_Str のパターン数を**表 3** に示す．ケース 1 と同様の場合分けに加え，さらに未入力文字数の違いごとに場合分けされる．

(b-3) (b-1)，(b-2) の一般化　未入力文字数を f，不一致文字数を g，⟨[A-D][1-9]⟩ の文字数を h とおく．また，認証文字列 Auth_Str に含まれる時空間文字の数を n，不一致許容数を tn，認証点の候補数を cn，認証動作のパターン数を en とおく．このとき，表 2，表 3 の ⟨[A-D][0-9]⟩ のパターン数は ${}_n\mathrm{C}_{n-g+h}$，⟨[A-D][1-9]⟩ のパターン数は ${}_{n-g+h}\mathrm{C}_h \times (en-1)^h$，⟨[E-Z][0-9]⟩ のパターン数は ${}_{cn-n}\mathrm{C}_{g-h-f} \times (en-1)^{g-h-f}$ となる．表 2，表 3 の各行は独立の事象である．また，表の各行の場合の数は，⟨[A-D][0-9]⟩ のパターン数 ×⟨[A-D][1-9]⟩ のパターン数 ×⟨[E-Z][0-9]⟩ のパターン数により求められる．不一致許容数が tn の場合の，認証に成功する入力文字列 Input のパターン数は，未入力文字数 f が 0 から tn の場合，不一致文字数 g が f から tn の場合，⟨[A-D][1-9]⟩ の文字数 h が 0 から g の場合の，Input のパターン数を合計したものとなる．

c　入力文字列一致率

(a-3)，(b-3) より，入力文字列一致率 pt_i は以下の式により求められる．なお，攻撃者は i 回目の認証試行において，それ以前に試みた入力文字列は試みないと仮定し，分母から $i-1$ を減じている．入力文字列一致率に基づき，式 7 を用いて FAR が計算可能になる（便宜上，$0^0 = 1$ とする）．

$$pt_i = \frac{\sum_{f=0}^{tn} \sum_{g=f}^{tn} \sum_{h=0}^{g} {}_n\mathrm{C}_{n-g+h}\ {}_{n-g+h}\mathrm{C}_h (en-1)^h\ {}_{cn-n}\mathrm{C}_{g-h-f}\, en^{g-h-f}}{\sum_{j=0}^{tn} {}_{cn}\mathrm{C}_{n-j}\, en^{n-j} - i + 1} \quad (9)$$

4. 実　　験

実験の目的は，3 章で説明した方法に基づき，ボタン押下に基づく認証動作を用いた場合の安全性を評価することである．まず，要素入力失敗率，認証点候補数と認証点マージンとの関係を実験により確かめた．次に，その結果に基づき，認証時間を 100 秒，試行可能回数を 2 回（すなわち入力文字列 Input の入力失敗を 1 回まで認める）として，文字列入力失敗率，本人拒否率（FRR），他人受入率（FAR）を計算した．実験では，測定誤差の影響を最小限にするために，測定誤差が最も小さい時間認証点 TP を用いた．

以下の手順により実験を行い，$tMargin_p$，$tMargin_a$ と要素入力失敗率との関係を調べた．

ステップ 1　被験者は 1 秒から 30 秒のうちの任意の移動時間 2 つを，時間認証点として設定する．

ステップ 2　被験者は移動しながら，ステップ 1 で設定した認証点でボタンを押す．

ステップ 3　ステップ 2 を 10 回繰り返す．

表 4 ボタン押下に基づく認証動作を用いた場合の要素入力失敗率と認証点候補数
Table 4 Element Input Failure Rate and Number of Authentication Point Candidate When Using Pushing Button Authentication Action.

$tMargin_a$(秒)	要素入力失敗率 (%)	認証点候補数
1	7.5	101
2	4.5	51
3	4	34
4	2.5	26

表 5 ボタン押下に基づく認証動作を用いた場合の文字列入力失敗率，FRR，FAR
Table 5 String Input Failure Rate, FRR, and FAR When Using Pushing Button Authentication Action.

時空間文字数	4			5		
不一致許容数	入力文字列失敗率 (%)	FRR(%)	FAR(%)	入力文字列失敗率 (%)	FRR(%)	FAR(%)
0	26.7906	1.9229	0	32.2813	3.364	0
1	3.047	0.0028	0.0185	4.8278	0.0113	0.0012
2	0.1593	0	1.3547	0.3758	0	0.1127
3	0.0032	0	27.7294	0.0149	0	3.6149
4				0.0002	0	40.1057
5						

被験者は 10 人，移動手段は徒歩とし，被験者は PDA を手に持ちながら直線の道路上で移動を繰り返した．移動時間 *etime* は PDA により被験者に提示し，タッチパネル機能を持つ液晶画面に表示されている GUI ボタンを認証点で押すこととした．

表 4 に $tMargin_a$ と要素入力失敗率，認証点候補数（認証時間を 100 秒とし，式 8 より計算）との関係を示す．認証点 TP よりも手前でボタンを押した被験者はいなかったため，$tMargin_p$ は 0 秒とした．ボタンを押したときの時空間情報 *spctmp* が $\mathrm{TP} - tMargin_p \leq spctmp < \mathrm{TP} + tMargin_a$ を満たしていないならば，*spctmp OUT* TP と判定され，要素入力失敗とみなされる．実験に用いた PDA は 1 秒未満の時間を計測できなかったため，$tMargin_a$ を 1 秒間隔としている．

$tMargin_a = 1$ 秒とすると，文字列入力失敗率と FAR をバランスよく抑えることができた．**表 5** に $tMargin_a = 1$ 秒の場合の文字列入力失敗率，FRR，FAR を示す．時空間文字を 4 つ，不一致許容数を 1 つとすると，文字列入力失敗率は 3.0%，FAR は 0.019%となる．時空間文字を 5 つ，不一致許容数を 1 つとすると，文字列入力失敗率は 4.8%となり利便性が低下するが，FAR は 0.0012%となり，安全性が高まる．なお，この場合の FRR は 0.2%であるが，試行可能回数を 3 回にすると，FAR が 0.0018%と少し高くなるが，FRR は 0.01%と非常に低くすることができる．

5. ま と め

本稿では，時空間情報と動作を組み合わせた認証方法を提案した．また，部分一致認証を提案し，時空間情報と動作に基づく認証方法の安全性の評価方法を明らかにするとともに，実験によりボタン押下を認証動作として用いた場合の安全性を確かめた．今後は，本稿で取り上げていない認証動作について，利便性や安全性を評価することを予定している．

参 考 文 献

1) 石原　進，太田雅敏，行方エリキ，水野忠則：端末自体の動きを用いた携帯端末向け個人認証，情報処理学会論文誌， Vol.46, No.12, pp.2997–3007 (2005).
2) Just, M. and Oorschot, P.: Addressing the Problem of Undetected Signature Key Compromise, *Proc. Network and Distributed System Security Symposium* (1999).
3) Matsumoto, T., Matsumoto, H., Yamada, K. and Hoshino, S.: Impact of Artificial“Gummy” Fingers on Fingerprint Systems, *Proc. Optical Security and Counterfeit Deterrence Techniques IV*, Vol.4677, pp.275–289 (2002).
4) 杉浦一成，槇原　靖，八木康史：全方位カメラを用いた複数方向の観測による歩容認証，情報処理学会論文誌: コンピュータビジョンとイメージメディア， Vol.1, No.2, pp. 76–85 (2008).
5) 鈴木雅貴，宇根正志：生体認証システムの脆弱性の分析と生体検知技術の研究動向，金融研究， Vol.28, No.3, pp.69–106 (2009).
6) Weaver, A.: Biometric Authentication, *IEEE Computer*, Vol.39, No.2, pp.96–97 (2006).
7) 読売新聞：「生体認証」破り入国，朝刊, 1 月 1 日 (2009).