サイバーデザインデータベース

木谷紀子
法政大学大学院ITプロフェッショナルコース
it013312@itpc.i.hosei.ac.jp

國井利泰
法政大学大学院ITプロフェッショナルコース
tosi@kunii.com; http://www.kunii.com/

要旨：ユーザの嗜好に合わせてデザインした実世界の物体を、サイバースペース上で仮想的にパーツセルに分割することにより、再利用可能なデータベース化する方法について、柔軟かつ明確に定義可能な新しい方法を開発した。分割する際のパーツセルの接着情報をデザイン情報としてデータベースに保存、蓄積し、かつパーツセルのデザインにホモトピー同等の関係が成立する様にすることにより、繰り返し効率良くデザインする事を可能にした。デザインデータを基にセル理論に基づくセルモデルにより、デザインの再利用、再設計が自由に可能となるようなデザインデータベースの構築が可能である事を示す。具体的なデザイン例として、説明の簡略化のために、日用品の代表的な存在でかつデザインが複雑でないバッグを対象とした。

Cyber Design Databases

Noriko Kitani
IT Professional Course, Graduate School,
Hosei University
it013312@itpc.i.hosei.ac.jp

Tosiyasu L. Kunii
IT Professional Course, Graduate School,
Hosei University
tosi@kunii.com; http://www.kunii.com/

ABSTRACT　A new flexible and well defined method was developed to turn objects in the real world, designed to satisfy users' taste, into reusable design resources in cyber worlds by virtually decomposing the original design into parts.　We show that we can repeat design processes efficiently by storing the information on part cell attachment as design information as well as by making part cell design processes homotopically equivalent.　We then show the possibility of a new architecture of design database management systems to support flexible design and redesign.　To demonstrate the power of the new method, bag design is selected as a simple and popular case.

1．緒言

サイバーワールドの一部であるサイバーデザイン空間にセルモデルを導入する事により、従来集合論レベルに限定された定義しか出来なかったデータベース間の関係、データベース内の関係を、ホモトピー理論、接着関数等を用いて動的に構成する事により、セルデータベースを構築する。[1]セルデータベースを元にサイバーデザインデータベースを構築する事でデザインの再利用、再設計を可能にする事を考える。

バッグは、デザイナーが紙と鉛筆を使ってデザイン画を描き、そのデザイン画を元に職人が型紙を作成し、生地をその型紙に合わせて、裁断し、ミシンをかけ、部品の取り付け等を行って作るのが一般的である。

完成した作品が1回でデザイナーのイメージ通りになるのは、上級の職人でさえ難しく、通常は、デザイナーと職人の意識合わせが何回か繰り返され、イメージ通りのバッグが製作される。そこにはデザイナーと職人との間にイメージの違いが存在する。デザイナーのデザイン画を元に各パーツをセルに見たて、分割していく事で、分割手順を保存し、そのデザイン情報及び分割情報（手順）を利用する事で組み立てのTry&Eroor を減らせないかを考える。

パーツが与えられ、ばらばらの状態からバッグを製作する場合、デザイン手順が保存されていなければ、各パーツを最初から試行錯誤しながら組み立てて行く事になる。

サイバーデザイン空間にセルモデルを導入し、サイバーデザインデータベースにデザイン情報を保存していく事による優位性も併せて示す。

2．リレーショナルデータベースとセルデータベース

サイバーデザインデータベースの構築にあたり、現在最も一般的に使用されているリレーショナルデータベースとセルデータベースを比較した。

リレーショナルデータベースは、データ同士の関係を表構造として持ち、その関係の組を集合として扱い、それらに演算を施すことによって情報を表現している。この構造は表として表現され、表の事をしばしばリレーションと呼ぶ。それぞれの表は，行とカラムから構成されている。

リレーショナルデータベースは変化が少なく、表形式で表現出来るもの、又決まった属性で表現出来るものに対しては非常に有効であるが、元々が集合論に基づいている事から、データの表現力は非常に弱い。

デザイン空間では、様々なデザインのバッグが入力され、修正、追加といった動的要素を多分に含む可能性がある事、またリレーショナルデータベースはデータ管理者による一元正規化を想定している事からリレーショナルデータベースを用いてデザイン空間を構築する事は、非常に難しい。

これに対し、セルデータベースは、情報を個々のセルに見たて、情報間の関係をセル同士の接着関数として保存し、接着前後でホモトピー同値関係を成立させる事で情報の再利用が可能となっている。[1]

このセルデータベースを元にサイバーデザインデータベースを構築する事で、動的にデザイン空間を構成出来る事を示す。

動的要素を含む情報をデータベース化する場合において、全ての属性を事前に定義付ける事は不可能である為、セルデータベースでは、リレーション間の関係は、リレーショナルデータベースの join オペレーションを一般化したものである equivalence relation を用いる。equivalence relation とは、同値関係の事であり、以下の関係を満たす場合を言う。[1]

反射性：if $(\forall x \in X)$ [xRx]

対称性：if $(\forall x, y \in X)$ [xRy ⇒ yRx]

推移性：if $(\forall x, y, z \in X)$ [[xRy ⇒ yRz] ⇒ xRz]

同値関係という非常に自由度を持った関係を用いる事で、サイバーデザインデータベースといった動的要素を多分に含むデータベースにも対応が可能となる。[1]

セルデータベースは、変化としてのセル結合とセル分解を許容する一方、不変物としてセルの次元及び接着可能性を維持している。[1]

3．ホモトピー理論

以下にホモトピーに理論に関する定義を示す。

$H: X \times I \to Y$

$(\forall x \in X)\ (H(x, 0) = f(x)$ かつ $H(x, 1) = g(x))$,

$(\forall a \in A, \forall t \in I)\ (H(a, t) = f(a) = g(a))$.

f は A に関して homotopic であり、

$f \simeq g$ (rel A)と記述する。

サイバー空間は様々な操作やプロセスによる変更を受ける。その変化する過程はホモトピー関数として保存される。分解の逆操作である結合が可能となるように、すなわち逆関数を考える事が出来るように、セル分解を homotopic に行う。この homotopic 不変性は、セル操作によって動的に変形されるセル情報を再利用可能にする点から非常に重要である。[1]

サイバーデザインデータベースでは、個々のパーツ及び属性をセルに見たて、その分解情報を保存する事で、ホモトピー同値関係を保持している。

4. サイバーデザインデータベース

デザイナーが自由にデザインしたバッグをサイバーワールド上にて認識し、個々のパーツをセルに見たて分解し、分解する上での接着情報及びセルに対応付けたデザイン情報を保存する事により、デザインの再現、再利用が可能になるサイバーデザインデータベースを構築する。

サイバーデザインデータベース構築にあたっての前提条件を以下に示す。

① カバンのデザインをデザイナー（ユーザ）が自由に行える。

② 実際の形状情報については、手書きソフトを用いて自由に手書きで行い、形状認識ソフトを用いて認識する。

③ 色、付属品等の細かい指定は、別にデザイナーの入力によって行う事とする。

④ デザイン認識されたバッグは、最大限に細かいパーツに分解する事とする。

データベースを構築して行く場合の例として、内部ポケット等を考えない、取っ手、チャック付きポケットから構成されるシンプルなバッグを考える。

デザイナーは形状情報と別に身ごろの色･素材、取っ手の色･素材、チャックの種類、寸法等の情報を入力しておく必要がある。

入力方法としては、対話形式又はユーザ入力形式とし、その情報は前もってサイバーデザインデータベースに蓄積しておく。

サイバーデザインデータベースの構築を以下の手順で行う。

Ⅰ. デザインされたバッグとセルとの対応付け

Ⅱ. 色、素材情報の入力及びデータベースへの情報保存

Ⅲ. 各パーツセルへの分解及びデータベースへの分解手順の保存

Ⅰ. デザインされたバッグとセルとの対応付け

デザインされたバッグに対して以下の図1から図3に示すようにセルと対応を取る。

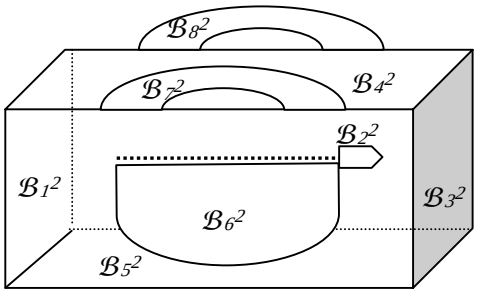

図1　デザインされたバッグの各面への対応

合計8の面が存在する。

$X^2 = \{\mathcal{B}_a^2 | a=1 \sim 8\}$

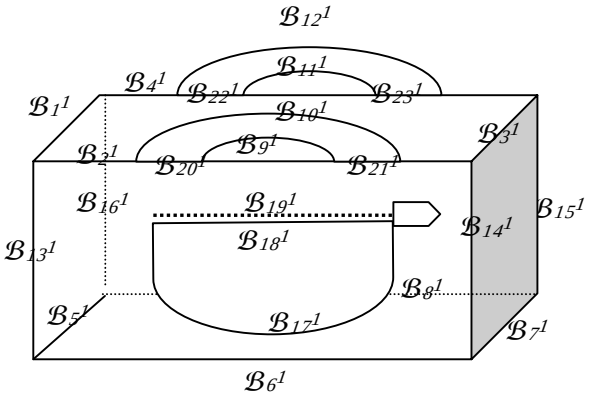

図2　デザインされたバッグの各稜線への対応

合計23の稜線が存在する。

$X^1 = \{\mathcal{B}_a^1 | a=1\sim23\}$

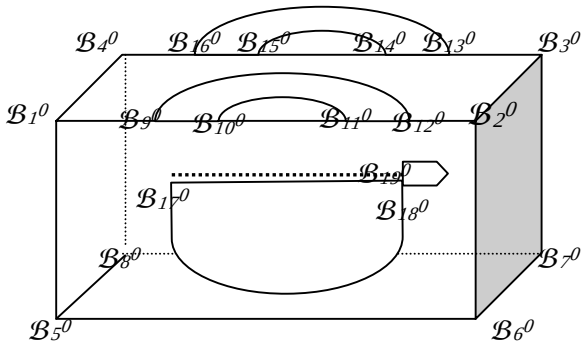

図3　デザインされたバッグの各頂点への対応

合計19の頂点が存在する。

$X^0 = \{\mathcal{B}_a^0 | a=1\sim19\}$

Ⅱ. 色、素材情報等の入力及びデータベースへの情報保存

バッグをデザインするには、デザイナーがデザインしたいバックの色情報、部品情報を事前に入力して、セルとの関係において、扱えなければならない。

ここで色情報をどのように扱うかを考えてみる。1次元の線は面積を持たないと考えられるが、それ等を非常に細い糸及びチャックと見なす事により、1次元のセルにも色情報を与える事とする。

また、2次元の面については、対応する個々の2次元セルが色情報を持つとする。

今回、例として取り上げたバッグは、「身ごろ部分の素材は牛革（カーフ）、色は、身ごろ及び底の部分は赤、取っ手の部分はベージュ、チャックは金色、糸の色は身ごろ部分と同様、ポケットの色は黒」であると仮定する。

表1、表2に示すように、1次元セル（糸の色及びチャック）及び2次元セル（面）の色情報をデータベースに保存する。

表1　1次元パーツ色 DB($\mathcal{B}_{dbcolor}^1$)

X^1	色
$\mathcal{B}_1^1$	赤
$\mathcal{B}_2^1$	赤
$\mathcal{B}_3^1$	赤
$\mathcal{B}_4^1$	赤
$\mathcal{B}_5^1$	赤
$\mathcal{B}_6^1$	赤
$\mathcal{B}_7^1$	赤
$\mathcal{B}_8^1$	赤
$\mathcal{B}_9^1$	ベージュ
$\mathcal{B}_{10}^1$	ベージュ
$\mathcal{B}_{11}^1$	ベージュ
$\mathcal{B}_{12}^1$	ベージュ
$\mathcal{B}_{13}^1$	赤
$\mathcal{B}_{14}^1$	赤
$\mathcal{B}_{15}^1$	赤
$\mathcal{B}_{16}^1$	赤
$\mathcal{B}_{17}^1$	黒
$\mathcal{B}_{18}^1$	黒
$\mathcal{B}_{19}^1$	金

表2　2次元パーツ色 DB($\mathcal{B}_{dbcolor}^2$)

X^2	色
$\mathcal{B}_1^2$	赤
$\mathcal{B}_2^2$	赤
$\mathcal{B}_3^2$	赤
$\mathcal{B}_4^2$	赤
$\mathcal{B}_5^2$	赤
$\mathcal{B}_6^2$	黒
$\mathcal{B}_7^2$	ベージュ
$\mathcal{B}_8^2$	ベージュ

素材情報も色情報と同様に与える事が出来る。

（表の詳細は省略する）

バッグをデザインする場合を考えると、色情報の他に、素材情報、チャックの種類、Nurbs情報、寸法情報、位置情報等が考えられる。

表3に示すように、Nurbs 情報（曲線情報）をデータベースに保存する。例に挙げたバッグで考えた場合、取っ手部分のカーブ（$\mathcal{B}_9^1$, $\mathcal{B}_{10}^1$, $\mathcal{B}_{11}^1$, $\mathcal{B}_{12}^1$）及びポケットの底部分（$\mathcal{B}_{17}^1$）のカーブがそれに該当する。（Nurbs曲線の詳細は省略する）

従来の CAD システムでは、曲線情報を座標として持っていた為、サイズの変更、多様な細部の製品展開が困難であった。セル理論を応用したデータベースでは、個々の曲線情報を0次元セルとその間の Nurbs

曲線に分解して情報を保持する為、保持すべき情報を非常にフレキシブルにする事が可能となる。

表 3　1次元 NurbsDB($\mathcal{B}_{db\,nurbs}^{1}$)

$\underline{X^1}$	Nurbs 曲線
$\underline{\mathcal{B}_9^1}$	Nurbs 曲線1
$\underline{\mathcal{B}_{10}^1}$	Nurbs 曲線2
$\underline{\mathcal{B}_{11}^1}$	Nurbs 曲線1
$\underline{\mathcal{B}_{12}^1}$	Nurbs 曲線2
$\underline{\mathcal{B}_{17}^1}$	Nurbs 曲線3

また位置情報は、ここでは簡略化の為、座標で与える事とする。位置情報データベース（一部）を以下の表4に示す。

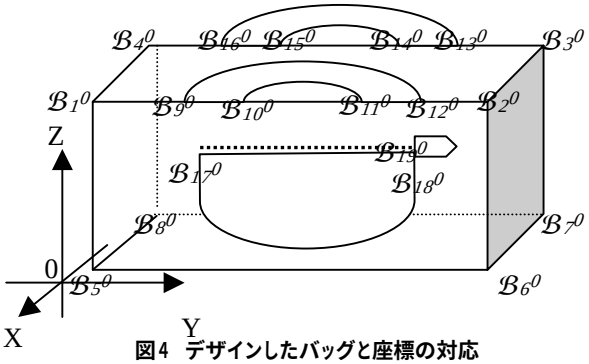

図4　デザインしたバッグと座標の対応

表4　0次元位置情報 DB($\mathcal{B}_{db\,位置}^{1}$)

$\underline{X^0}$	位置
$\underline{\mathcal{B}_9^0}$	(X_9,Y_9,Z_9)
$\underline{\mathcal{B}_{10}^0}$	(X_{10},Y_{10},Z_{10})
$\underline{\mathcal{B}_{11}^0}$	(X_{11},Y_{11},Z_{11})
$\underline{\mathcal{B}_{12}^0}$	(X_{12},Y_{12},Z_{12})

III. 各パーツセルへの分解及びデータベースへの分解手順の保存

前提条件にて設定した様に、デザインされたバッグを最大限に分解していく。バックは図1に示すように、8個の面から構成されている。

X^2 バッグ$=\mathcal{B}_1^2 \sqcup \mathcal{B}_2^2 \sqcup \mathcal{B}_3^2 \sqcup \mathcal{B}_4^2 \sqcup \mathcal{B}_5^2 \sqcup \mathcal{B}_6^2 \sqcup \mathcal{B}_7^2 \sqcup \mathcal{B}_8^2$

分解の順序として以下を考える。

$\mathcal{B}_1^2 \rightarrow \mathcal{B}_2^2 \rightarrow \mathcal{B}_3^2 \rightarrow \mathcal{B}_4^2 \rightarrow \mathcal{B}_5^2 \rightarrow \mathcal{B}_6^2 \rightarrow \mathcal{B}_7^2 \rightarrow \mathcal{B}_8^2$

上記順序に従ってデザインされたバッグのセル分解を行っていく事とする。

以下表5に示すように、接着情報をデータベースに保存し、各々の接着関数を以下表5のように対応付ける。

表5　2次元面の接着情報 DB($\mathcal{B}_{db\,attach}^{2}$)

$\underline{X^2}$	接着面	接着関数
$\underline{\mathcal{B}_1^2}$	$\mathcal{B}_2^2$	F_1
$\underline{\mathcal{B}_1^2}$	$\mathcal{B}_4^2$	F_2
$\underline{\mathcal{B}_1^2}$	$\mathcal{B}_5^2$	F_3
$\underline{\mathcal{B}_2^2}$	$\mathcal{B}_3^2$	F_4
$\underline{\mathcal{B}_2^2}$	$\mathcal{B}_5^2$	F_5
$\underline{\mathcal{B}_2^2}$	$\mathcal{B}_6^2$	F_6
$\underline{\mathcal{B}_2^2}$	$\mathcal{B}_7^2$	F_7
$\underline{\mathcal{B}_3^2}$	$\mathcal{B}_4^2$	F_8
$\underline{\mathcal{B}_3^2}$	$\mathcal{B}_5^2$	F_9
$\underline{\mathcal{B}_4^2}$	$\mathcal{B}_5^2$	F_{10}
$\underline{\mathcal{B}_4^2}$	$\mathcal{B}_8^2$	F_{11}

①　まち部分 ($\mathcal{B}_1^2$) の取り出し

上記表5に示すように、まち部分（$\mathcal{B}_1^2$）は、3つの2次元セル（$\mathcal{B}_2^2$, $\mathcal{B}_4^2$, $\mathcal{B}_5^2$）と0次元の点（$\mathcal{B}_1^0$,$\mathcal{B}_4^0$,$\mathcal{B}_5^0$,$\mathcal{B}_8^0$）及び1次元の線（$\mathcal{B}_5^1$, $\mathcal{B}_{13}^1$, $\mathcal{B}_{16}^1$）で接着している。表6に示すように、この接着情報をデータベースに保存する。

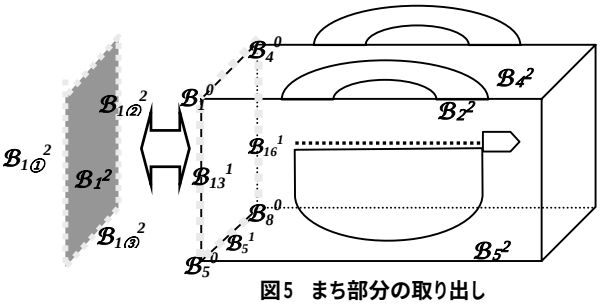

図5　まち部分の取り出し

表6　2次元接着情報 DB ($\mathcal{B}_{dbattach}^{2}$)

$\underline{X^2}$	接着関数	0 次元	1次元
$\underline{\mathcal{B}_1^2}$	F_1:$\mathcal{B}_{1①}^{2}\rightarrow\mathcal{B}_2^2$	$\mathcal{B}_1^0,\mathcal{B}_5^0$	$\mathcal{B}_{13}^1$
$\underline{\mathcal{B}_1^2}$	F_2:$\mathcal{B}_{1②}^{2}\rightarrow\mathcal{B}_4^2$	$\mathcal{B}_4^0,\mathcal{B}_8^0$	$\mathcal{B}_{16}^1$
$\underline{\mathcal{B}_1^2}$	F_3:$\mathcal{B}_{1③}^{2}\rightarrow\mathcal{B}_5^2$	$\mathcal{B}_5^0,\mathcal{B}_8^0$	$\mathcal{B}_5^1$

($\mathcal{B}_{1①}^{2}$、$\mathcal{B}_{1②}^{2}$、$\mathcal{B}_{1③}^{2}$については図5参照)

②　取っ手 ($\mathcal{B}_1^2$) と身ごろ ($\mathcal{B}_2^2$) の分解

($\mathcal{B}_8^2$ は同様)

取っ手と身ごろは、4つの0次元セル($\mathcal{B}_9^0$,$\mathcal{B}_{10}^0$,$\mathcal{B}_{11}^0$,$\mathcal{B}_{12}^0$)と身ごろの1次元セル($\mathcal{B}_{16}^1$)上において接着されている。(図6参照)

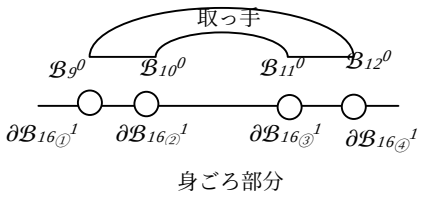

図6　取っ手と身ごろの接着点に対する対応

表7に示すように、取っ手と身ごろの接着情報をデータベースに保存する。

表7　0次元接着関係 DB ($\mathcal{B}_{dbattach}^{0}$)

$\underline{X^0}$	関数	$\partial\mathcal{B}_{16}^1$
$\underline{\mathcal{B}_9^0}$	$F_{7①}$:$\partial\mathcal{B}_{16①}^1\rightarrow\ \mathcal{B}_9^0$	$\partial\mathcal{B}_{16①}^1$
$\underline{\mathcal{B}_{10}^0}$	$F_{7②}$:$\partial\mathcal{B}_{16②}^1\rightarrow\ \mathcal{B}_{10}^0$	$\partial\mathcal{B}_{16②}^1$
$\underline{\mathcal{B}_{11}^0}$	$F_{7③}$:$\partial\mathcal{B}_{16③}^1\rightarrow\ \mathcal{B}_{11}^0$	$\partial\mathcal{B}_{16③}^1$
$\underline{\mathcal{B}_{12}^0}$	$F_{7④}$:$\partial\mathcal{B}_{16④}^1\rightarrow\ \mathcal{B}_{12}^0$	$\partial\mathcal{B}_{16④}^1$

取っ手と身ごろの接着の対応をとって、その接着関係を保存する事により、関数 $F_{7①}$～$F_{7④}$は、逆関数を考える事が出来、糸を外す前の状態と外した後の状態は、ホモトピー同等になっている。

また身ごろの境界線上に存在する取っ手との境界点に座標を用いた位置情報を与える事で、取っ手と身ごろの接着状態を忠実に再現する事が可能となる。

ここで、関数 $F_{7①}$～$F_{7④}$の操作内容を具体的に考えると、バッグは糸を使って縫い合わせてあるので、糸を外すという操作になる。関数 $F_{7①}$～$F_{7④}$の詳細を以下表8に示すように保存する。

表8　0次元接着関数 DB($\mathcal{B}_{dbattach}^{0}$)

$\underline{X^0}$	接着関数	関数詳細
$\underline{\mathcal{B}_9^0}$	$F_{7①}$:$\partial\mathcal{B}_{16①}^1\rightarrow\ \mathcal{B}_9^0$	糸を外す
$\underline{\mathcal{B}_{10}^0}$	$F_{7②}$:$\partial\mathcal{B}_{16②}^1\rightarrow\ \mathcal{B}_{10}^0$	糸を外す
$\underline{\mathcal{B}_{11}^0}$	$F_{7③}$:$\partial\mathcal{B}_{16③}^1\rightarrow\ \mathcal{B}_{11}^0$	糸を外す
$\underline{\mathcal{B}_{12}^0}$	$F_{7④}$:$\partial\mathcal{B}_{16④}^1\rightarrow\ \mathcal{B}_{12}^0$	糸を外す

次に分解された2次元セルに対して0次元まで次元を下げ、その分解情報をデータベースに保存していく。例として取っ手 ($\mathcal{B}_7^2$) を考える。

③　取っ手部分の分解 (2次元→1次元→0次元)

取っ手部分は、2次元のセルから構成されているとみなす事が出来る。

$X^2_{取っ手}=\{\mathcal{B}_7^2\}$

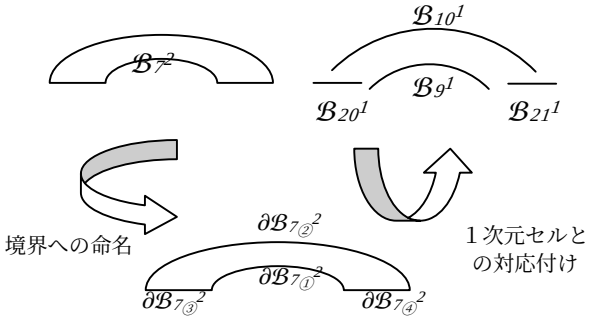

図7　取っ手部分の分解1

表9に示すように、取っ手の分解情報をデータベースに保存する。

表9　1次元接着関係 DB ($\mathcal{B}_{dbattach}^{1}$)

$\underline{X^1}$	関数	$\partial\mathcal{B}_7^2$
$\underline{\mathcal{B}_9^1}$	$G_{7①}$:$\partial\mathcal{B}_{7①}^2\rightarrow\ \mathcal{B}_9^0$	$\partial\mathcal{B}_{7①}^2$
$\underline{\mathcal{B}_{10}^1}$	$G_{7②}$:$\partial\mathcal{B}_{7②}^2\rightarrow\ \mathcal{B}_{10}^0$	$\partial\mathcal{B}_{7②}^2$
$\underline{\mathcal{B}_{20}^1}$	$G_{7③}$:$\partial\mathcal{B}_{7③}^2\rightarrow\ \mathcal{B}_{20}^0$	$\partial\mathcal{B}_{7③}^2$
$\underline{\mathcal{B}_{21}^1}$	$G_{7④}$:$\partial\mathcal{B}_{7④}^2\rightarrow\ \mathcal{B}_{21}^0$	$\partial\mathcal{B}_{7④}^2$

更に、分解された取っ手部分の1次元セル（$\mathcal{B}_9^1$,$\mathcal{B}_{10}^1$,$\mathcal{B}_{20}^1$,$\mathcal{B}_{21}^1$）を4つの0次元セル($\mathcal{B}_9^0$,$\mathcal{B}_{10}^0$,$\mathcal{B}_{20}^0$,$\mathcal{B}_{21}^0$)に分解する。

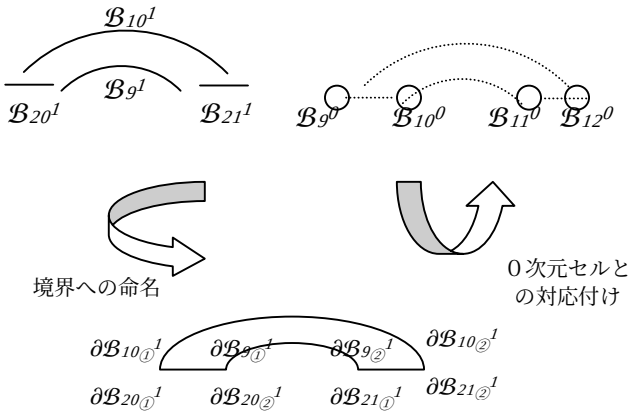

図8 取っ手部分の分解2

表10 0次元接着関係DB ($\mathcal{B}_{dbattach}^0$)

$\underline{X^0}$	関数	$\partial\mathcal{B}_7^1$
$\underline{\mathcal{B}_9^0}$	$H_{7①}$:$\partial\mathcal{B}_{10①}^1 \rightarrow \mathcal{B}_9^0$	$\partial\mathcal{B}_{10①}^1$
$\underline{\mathcal{B}_9^0}$	$H_{7②}$:$\partial\mathcal{B}_{20①}^1 \rightarrow \mathcal{B}_9^0$	$\partial\mathcal{B}_{20①}^1$
$\underline{\mathcal{B}_{10}^0}$	$H_{7③}$:$\partial\mathcal{B}_{9①}^1 \rightarrow \mathcal{B}_{10}^0$	$\partial\mathcal{B}_{9①}^1$
$\underline{\mathcal{B}_{10}^0}$	$H_{7④}$:$\partial\mathcal{B}_{20②}^1 \rightarrow \mathcal{B}_{10}^0$	$\partial\mathcal{B}_{20②}^1$
$\underline{\mathcal{B}_{11}^0}$	$H_{7⑤}$:$\partial\mathcal{B}_{9②}^1 \rightarrow \mathcal{B}_{11}^0$	$\partial\mathcal{B}_{9②}^1$
$\underline{\mathcal{B}_{11}^0}$	$H_{7⑥}$:$\partial\mathcal{B}_{21①}^1 \rightarrow \mathcal{B}_{11}^0$	$\partial\mathcal{B}_{21①}^1$
$\underline{\mathcal{B}_{12}^0}$	$H_{7⑦}$:$\partial\mathcal{B}_{10②}^1 \rightarrow \mathcal{B}_{12}^0$	$\partial\mathcal{B}_{10②}^1$
$\underline{\mathcal{B}_{12}^0}$	$H_{7⑧}$:$\partial\mathcal{B}_{21②}^1 \rightarrow \mathcal{B}_{12}^0$	$\partial\mathcal{B}_{21②}^1$

以上により2次元セルの取っ手部分が0次元セルまで分解された。
上記手順を繰り返し、デザインされたバッグを最大限に細かいパーツに分解して行く。

5．デザインの再現

次に保存されたデータベースの情報をもとに、デザインされたバッグを忠実に再現していく。
再現する際には、データベースの中を目的のセルに着目して検索する。

例えば、取っ手部分を再構成するには、取っ手部分の0次元セル $\mathcal{B}_9^0$ で検索する。
検索した結果、以下の情報を得る事が出来る。

- 座標情報：(X_9,Y_9,Z_9)（表4より）
- 接着関数情報1：$\mathcal{B}_9^0$ は $\partial\mathcal{B}_{10①}^1$,$\partial\mathcal{B}_{20①}^1$ の点で取っ手を形成（表10より）
- 接着関数情報2：$\mathcal{B}_9^0$ は $\partial\mathcal{B}_{16①}^1$ の点で身ごろ部分と接着（表7より）

  その際の接着関数は（表8より）

  $F_{7①}$：$\partial\mathcal{B}_{16①}^1 \rightarrow \mathcal{B}_9^0$（糸を外す）

これらの情報をもとに、接着関数の逆関数を考え、構築、検索の過程を繰り返す事により、デザインデータベースに保存されたデザインを忠実に再構築していく。接着関数の逆関数は

$F_{7①}^{-1}$：$\mathcal{B}_9^0 \rightarrow \partial\mathcal{B}_{16①}^1$（糸で縫う）

となる。取っ手部分の再構築を以下の図9に示す。

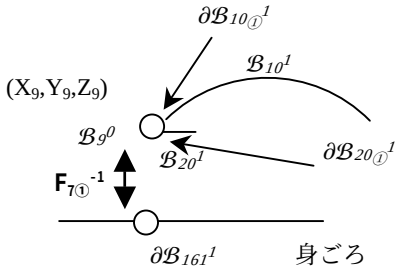

図9 取っ手部分の再構築

6．分解情報保存による優位性

分解情報をデータベースに保存する事により、同じデザインのものを忠実に再現出来るという利点と効率良く無駄なく再構築が出来るという利点がある。分解情報を保存する事による優位性を以下に示す。

N 個以下のパーツからバッグが構成されているとする。N 個以下のパーツがばらばらの状態で存在した場合、バッグをその状態から構成するには、順序を考慮に入れた場合、最大

$$A = {}_NC_N \times N! + {}_NC_{N-1} \times (N-1)!$$
$$+ \cdots\cdots + {}_NC_1 \times 1 + 1 \text{通り}$$

のパターンが存在する。

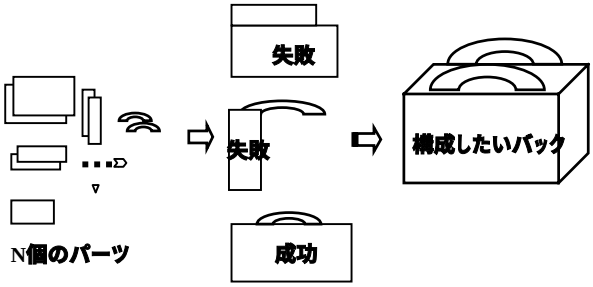

図10　分解情報未保存の場合

これに対し、N 個以下のパーツから構成されているバッグを考えた場合、構成された状態からデザインデータベースに保存されている各パーツへの対応を取る場合を考える。

デザインデータベースに N 個のパーツが保存されている場合、バッグを構成している N 個以下のパーツと対応付ける。その場合の数は、最大、$B = N \times N = N^2$ 通りのパターンが存在する。[2]

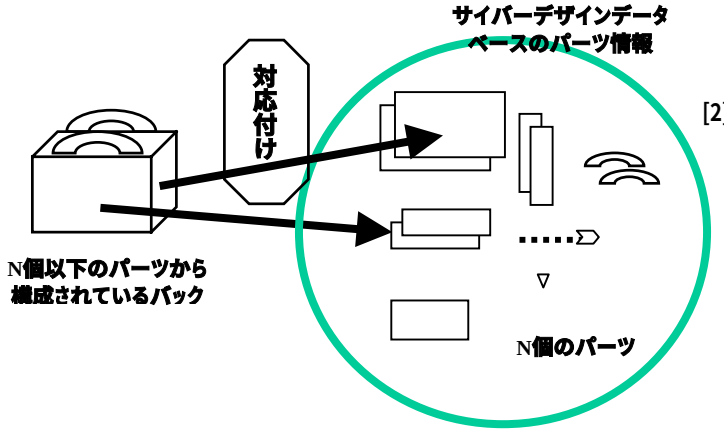

図11　分解情報が保存されている場合

N が大きい程、明らかに

A ＞ B が成立する。

例として挙げたバックは 8 の面から構成されているので、

$$A = {}_8C_8 \times 8! + {}_8C_{8-1} \times (8-1)! + {}_8C_{8-2} \times (8-2)!$$
$$+ {}_8C_{8-3} \times (8-3)! + {}_8C_{8-4} \times (8-4)! + {}_8C_{8-5} \times (8-5)!$$
$$+ {}_8C_{8-6} \times (8-6)! + {}_8C_{8-7} \times (8-7)! + {}_8C_{8-8} \times (8-8)!$$
$$= 109{,}601 \text{ 通り}$$
$$B = 8 \times 8 = 64 \text{ 通り}$$

明らかに A>B が成立する。

7. 参考文献

[1]T. L. Kunii and H. S. Kunii, “A Cellular Model for Information Systems on the Web - Integrating Local and Global Information”, Proceedings of 1999 International Symposium on Database Applications in Non-Traditional Environments (DANTE'99), November 28-30, 1999, Heian Shrine, Kyoto, Japan, Organized by Research Project on Advanced Databases, in cooperation with Information Processing Society of Japan, ACM Japan, ACM SIGMOD Japan, pp. 19-24, IEEE Computer Society Press, Los Alamitos, California, U. S. A.

[2] Tosiyasu L. Kunii,Tsukasa Noma, Kyujac Lee “ASSEMBLABILITY DISCRIMINATING METHOD AND ASSEMBLING SEQUENCE GENERATING METHOD”United States Patent