作図を目的とした摺合せ作業における認識の不一致を検出する手法の提案

A Method for Detecting Misunderstandings in Cooperative Drawing

海谷 治彦*

Summary. It is necessary for the cooperative workers to have mutual understanding with each other. But they sometimes can not realize that they do not have, hence the efficiency of the task goes down and inconsistent products are produced. In this paper, I introduce a method for finding symptoms of misunderstandings among the cooperative workers from the observational phenomena in their activities. This method is designed for the task such as developing diagrams by several workers. For validating the method, I designed a specific task, namely *negotiation task for software diagrams*, and had experiments of the task. This task is good both for studying misunderstandings and for having experiments, because it can not be simply shared among the workers and they should have a lot of observational phenomena such as drawing, pointing and talking during their activities. From the results of the experiments, I discuss advantages and limits of the method, and the computer support for the method.

1 はじめに

共同作業を通して，なんらかの生産物を作成する場合，他者との合意形成をもとにした意志決定，場合によっては折衝や妥協なども必要となる．しかし，合意形成や意志決定，折衝や妥協が正しく行なわれるためには，共に作業する他者との間に，作業対象のことがらについての認識が一致していることが前提となる．例えば，話題となっている事柄を，他者はどのように理解しているか，もしくは，話題としている事柄自身が作業者間で共有されているかなどが保証されていなければ，意味のある合意，意志決定，折衝などは期待できない．特に，ソフトウェア開発などに代表される決定の積み重ねを必要とする一連の作業では，初期の段階の問題点が，後段階の作業に対して致命的な打撃を与えることは良く知られている [1]．しかし，互いの認識が一致していないことを作業者自身が認識していないために，非効率的な作業や，互いに整合性のとれない生産物群などが発生する場合がある．

よって，共同作業の支援の一つとして，共通理解の不一致を検出する手法やツールが必要となる．その一例として，KJ 法などの図解法をもとにして，共通理解の一

* Haruhiko Kaiya, 北陸先端科学技術大学院大学 情報科学研究科

致/不一致を表示する方法などが提案されている [2], [3] . また , 設計作業履歴 [4] を , 作業の参加者や , 参加できなかった関係者が参照することで , 作業に関する認識の不一致を軽減することが可能であると思われる . しかし , これらの技術を適用するために従来の作業手順を変更することは , 実際には困難な場合が多い . そこで , 従来の作業方法をできる限り阻害しない方法で , 認識の不一致の発生を検出する方法が望ましいと思われる .

そこで本稿では , 作業中に外部から観測可能な活動をもとに , 理解の不一致の発生を検出するための方法論を提案する . 支援する作業としては , 図を分担して作成する作業に焦点を絞り , その作業中に話題となっている図の切り替わりに注目し , 理解の不一致の発生を検出する . 続く 2節では , 認識の不一致の発生原因に基づいて , その分類を行ない , 本稿で支援対象とする作業内容の位置付けを明らかにする . 3節では , 不一致の検出の手続きを紹介する . そして , 4節において , 実験的に設定した作業の分析から , 本稿の手法の有効性 , 妥当性を検討する . そして最後に , まとめと , 本方法の限界 , 計算機による支援の可能性などについて議論する .

2 認識の不一致の分類

議論の対象となる問題に関して , 認識の不一致が発生する原因は以下の 2 種類に分類できる .

1. 互いに持つ情報が異なる場合: 先生と生徒の関係に見られるように , 一方が一方的に知識を持っており , 他方にそれを伝達するという場合 . 全く新しい企画などを考える場合も , それぞれが事前の理解を持たないという点でこの場合に入る .

2. 同じ対象を異なる表現で理解する場合: 互いの立場の違いから同じ対象の違う側面を見ている場合など . 例えば , ソフトウェア設計では構造 , 機能 , 動作の複数の視点から設計対象を捉えることが多いため , それらの視点間の整合性をとることが重要となる . また , 要件定義の顧客と設計者などの持つ視点の差異なども同様である .

実際の作業における不一致は , 上記の両方の要因を含むと思われるが , 本稿では , 主に後者の要因が中心となるような不一致が発生しやすい作業の中で , 文章や図などの生産物を共同で作成する作業を対象とする . なぜなら , そのような作業は単純な情報伝達作業とは異なり , 実際に一致しているかどうかを作業者自身が認識するのが困難だからである . また , 作業は , 特別なツールなどを利用しない一般的な作業形態である , 共有黒板などを利用し , 自由に会話を行ないながら進める形態とする .

3 不一致の検出

本稿の手法は作業中の以下の点に注目して , 認識の不一致が発生している部分を予測する .

- 議論対象となる図や資料は , 一連の作業の間に何回も切り替わる . 例えば , 会議などでは , 複数の図を書いたり , 資料や参考書を途中で参照することで , 議論の中心となる資料が刻々と変化する .

- ある図から他の図への議論の切り替わりは，特定の作業者が，新しい図の上の要素を指したり，書き足したりする操作で始まる．この切り替えを行なった作業者をイニシアティブと呼ぶことにする．
- ある図に関する議論が終了するまでに，それぞれの作業者が，色々な操作を図の要素に対して施す．場合によっては操作を行なわない作業者もいる．
- ある図に関する議論が終了した時に施された操作は，その議論の結論に結び付いている．

これらの点に注目し，以下の手順で，認識の不一致が発生していると思われる生産物の部分を予測する．

1. 図の切り替わりをもとに，作業時間をいくつかの区間に分割する．
2. 区間で議論対象となっている図のイニシアティブを認識する．
3. 区間の中での操作と操作者を認識する．
4. もし，イニシアティブ以外の作業者が全く操作を行なっていない場合，不一致が発生する危険が高いとする．
5. もし，最終操作がイニシアティブでない場合，不一致の発生は少ないとする．
6. もし，それぞれの作業者が同じ部分の操作を行なった場合，すなわち操作の追認を行なった場合，不一致が発生する危険は低いとする．

この手順は，ある区間内で主導権を持つイニシアティブだけが作業を進め，他の作業者の関与が少ない場合，共通の理解が得られていない危険が高いという仮説に基づいている．実際には，極端に小さい区間の統合や，大きい区間の分割を行なう．その場合，会議のメタ発言 [5] に注目したり，作業者の話者交替の分布 [6] に注目したりする．

4 実作業の分析を通した手法の評価

本節では，実験的に設定した摺合せ作業による作業プロトコルの分析を通して，本稿で提案する手法の有効性や問題点について議論する．

4.1 実験の目的と設定

複数の作業者が異なった視点から同一の対象に関しての生産物を作成し，その摺合せを行なう作業を以下のように実験的に設定した．具体的な作業内容は，2 人の被験者がソフトウェアの分析/設計法である OMT[7] の機能モデルもしくは動作モデルのどちらかを用いて酒問屋の業務の分析 [8] を個々に行ない，その摺合せを行なうことである．本作業は，以下に示すような点から，認識の不一致に関する研究を実験的に行なう場合に適している．

- 2節に示した不一致の分類の後者にあたる，同じ対象を異なる観点から取り組む要素が強い．
- 作業が作図であるため，分析者が観測可能な事象が豊富である．
- 作成する図の構文が明確なため，作業の分析が容易である．

実験者は，以下の順序で被験者に対して指示を行ない実験を進めた．

1. それぞれの被験者に自然言語で書かれた要件と簡単な構造モデルを渡し，機能モデルもしくは，動作モデルでの業務分析を行なうように指示する．機能モデルではデータフロー図 (DFD) を作図する作業が中心となり，動作モデルでは状態遷移図 (STD) を作図する作業が中心となる．要件は，20 文程度の曖昧な部分を多く含むものである．さらに，後日，自分の担当でない方の図を記述した被験者 (パートナー) との摺合せ作業 (会議) を行なうことを告知する．また，会議日までは，パートナーとは連絡をとらないように指示する．

2. 図 1に示すような形態で，2 つのモデルの摺合せ作業 (会議) を行なってもらう．被験者は，自分の書いてきた図，要件，教科書などを参照しながら議論してよいが，それぞれが準備してきた図を直接相手に見せることはせず，黒板に写しながら説明を行なう．また，摺合せの基準として，自分の書いた図の修正が最小となるように指示する．

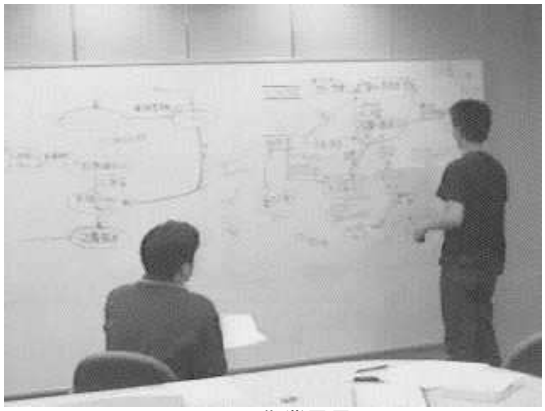


図 1. 作業風景

3. 摺合せ後に，被験者に，会議の議論を踏まえて，それぞれ個別に機能モデルと動作モデルの両方を記述するように指示する．この際，会議中のメモ，および他者の準備してきた図は実験者の方で回収し，被験者には参照させない．

摺り合せ会議後の作図は，記憶の劣化の影響を最小にするため，会議終了後，できるだけ早く，それぞれの被験者に記述させた．また，会議中に，他者の図を記憶することに専念させないために，他者の分担の図を会議後に記述する指示は会議の後に行なった．

4.2 実験の分析方法

分析では，それぞれの被験者が会議後に作成した図の差異をもとに認識の不一致の存在を確認し，不一致部分に関する会議中の区間が，3節の手続きに示されているような性質を持っているかどうかを確認する．図 2に，実験の手順と分析の手順を併せて示す．会議後に，機能モデルを担当した被験者 1 は，自分が前もって書いた DFD の改定版である DFD1' と，会議での理解をもとに STD1 を記述する．被験者 2 も同様である．そして分析者は被験者 1 の DFD1' と被験者 2 の DFD2，被験者 1 の

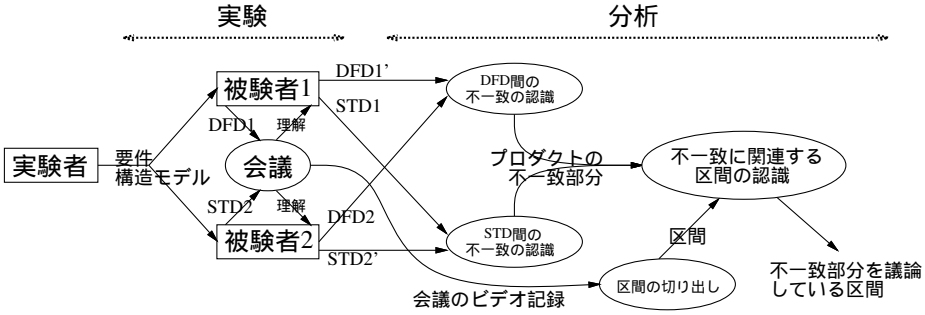


図 2. 実験と分析の手順

STD1 と被験者 2 の STD2' の差異をもとに認識の不一致を認識する．さらに，会議内容のビデオカメラによる記録を 3節の手続きに従い区間に分ける．そして認識の不一致に関する議論が行なわれている部分の活動のプロトコルが 3節の手続きに示す性質を持つかどうかを確認する．

実世界の作業では，摺合せ後に他者の分担部分まで記述することはほとんどなく，必要ならば自分の分担部分を修正する程度だと思われる．また，ソフトウェア工学的な観点からも上記の作業方法は現実的/理想的ではない．しかし，本稿で提案する手法の目的が，摺合せ会議を行なった後でも，なお内在する共同作業者間の認識の不一致を作業過程の情報から獲得しようとするものである．よって，あえて本実験では摺合せ後に他者の分担の図まで記述させることにした．

認識の不一致は会議後に作成された図の差異をもとに決定する．まず，図中の節や枝の数，そこについた注釈，節や枝の間の接続の違いを構文不一致とする．注釈の違いのうち，内容的に同一のものは差異とはしない．例えば，“数” と “数量” などの差異，複合データ構造を，そのまま注釈として記述してあるか個々のデータに展開してあるかなどは差異とはしない．さらに，いくつかの構文不一致がまとまって内容的な不一致を構成しているため，それを内容不一致とする．内容不一致の中でも接続の違いを含むものを重大な不一致とする．図 3に，構文不一致と内容不一致の例を示す．図中の 2 つのデータフロー図では，アクタからのデータの流れ先と，プロセスとデータストアの流れの方向が逆であるため，構文不一致は 2 であるが，あるデータフローをどのタイミングで保存するか，二次加工するかの違いと捉えると内容的には 1 つの不一致と考えられる場合がある．また図のトポロジが異なるため重大な不一致でもある．

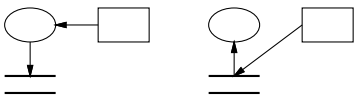


図 3. 構文不一致と内容不一致

4.3 結果と考察

被験者は本学の OMT の講義の合格者から募った．表 1に被験者と作業の概要を示す．被験者が注目している図の切り替わりを用いて分割した区間の平均時間は，面

識がある場合は若干長いが，それほど大きな差はない．また，平均区間長は，区間内の議論内容を概観するのに十分小さい長さであると思われる．

表 1. 被験者ペアの概要と作業時間

	被験者間の面識	会議時間 (分)	平均区間長 (分)
ペア 1	なし	140	1.7
ペア 2	あり	50	2.1

表 2, 3に，会議後にそれぞれに作成された生産物 (図) の規模と，それらの図の差異から見た認識の不一致の程度を示す．規模は節の数と枝の数のペアで示し，重大な不一致の数はカッコ内に付記している．また，それぞれの表中の被験者 1，被験者 2，DFD1', DFD2, STD1, STD2' などは，図 2内のそれと対応している．比較する図の間において，図の規模の差異はほとんどないが，かなりの数の構文的な差異が見られる．その多くは節や枝への注釈の違いなどであった．しかし，双方のペアにおいて，トポロジの違いを中心とした重大な認識の不一致が見られた．

表 2. 図の規模と認識の不一致の概要 (ペア 1)

	機能モデル	動作モデル
被験者 1	13 + 16 (DFD1')	7 + 10 (STD1)
被験者 2	13 + 16 (DFD2)	6 + 12 (STD2')
構文不一致数	16	6
内容不一致数	6 (2)	4 (0)

表 3. 図の規模と認識の不一致の概要 (ペア 2)

	機能モデル	動作モデル
被験者 1	15+20 (DFD1')	7+10 (STD1)
被験者 2	12+15 (DFD2)	7+10 (STD2')
構文不一致数	20	8
内容不一致数	6 (3)	3 (1)

本分析では，以下に示す図への操作に注目する．

point : 図中の特定の要素や部分を指などで指す

trace : 図中の複数の要素間の関係を指などでなぞる

draw : 図に新しい要素を書き加える

delete : 図の要素を削除する

point と trace の場合，発語や動作によって消したいことを示唆していることが明白な場合，そのことを注釈として付加している．そして，不一致部分に関連する会議中の区間の中に含まれる操作の列を調べることで，本稿で提案している方法に合致

しているかを調べる．以下に，それぞれのペアの代表的な不一致についての分析を例示する．

図 4にペア 1 に関する重大な不一致を示す．左側の図 DFD1' が被験者 1 の作成した図，右側の図 DFD2 が被験者 2 の作成した図である．ここでの不一致は，

データ C が，A と B のどちらで作成されるか?

ということが中心となっている．図 5, 6, 7にはその不一致部分に関する会議内の区間を示す．図 5, 6, 7のそれぞれには，その区間の開始時点の図，区間中に操作された部分を示す図，終了時点の図，区間内で行なわれた操作列を示した表からなる．開始時点の図と終了時点の図は，場合によっては省略してある．また，紙面の都合で，trace 以外の操作で連続して同一の操作者によって行なわれたものは，一行にまとめて表中に記述してある．例えば，図 6は会議開始 70 分から 71 分の間の区間に関する図の変化を示しており，この区間の一番最初では，この部分の担当者である被験者 1 が，図中の 1 と 2 のデータの流れをなぞっていることが記録されている．

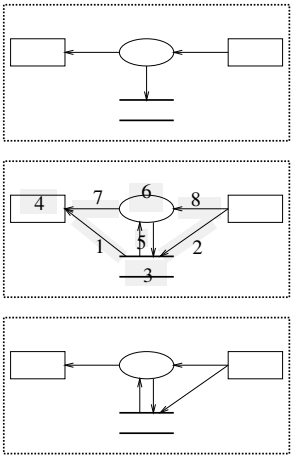
まず，図 5の区間においては，3節の方法によると，イニシアティブが被験者 1 となり，それ以外の作業者の操作も多く含まれているため，不一致の危険は少ないと判定できる．特に，図 5の操作列の 3, 4, 5 行目から，被験者 1 の提案の被験者 2 による追認があるため，共通理解が得られたと考えられ，その内容は被験者 2 の最終成果物 (図 4の DFD2) と合致している．しかし，最終操作者は，被験者 1 自身なので，この観点からは危険であると判定できる．図 6の区間からも，不一致の危険は少ないと判定でき，区間終了時点の図から，図 4の DFD2 に一致したと予想できる．図 7の区間では開始/終了の操作者の観点からは不一致が少ないと考えられるが，操作のほとんどが被験者 1 によって行なわれており，被験者 2 による追認もほとんど行なわれず，会議前の図 (図 4の DFD1') に似た図に戻っている．これより，作業中の作業者がお互いの操作を追認する行為がないことは，不一致の危険を示す指標となりうるが，イニシアティブは区間の開始を行なった作業者ではなく，より多くの操作を行なった作業者で代替した方が良いと思われる．また，図 4と図 5から，この部分の担当者である被験者 1 は，もともと自分の準備した図に固執する傾向があることがわかる．よって，操作列とその操作者だけではなく，事前に準備された図 (ここでは DFD1) との比較も加味して認識の不一致の発生を予測すべきであると思われる．

次に，図 8にペア 2 に関する重大な不一致を示す．ここでの不一致は，

プロセス A の出力データは，B と C のどちらに流すべきか?
データ b はどこで生成されるのか?

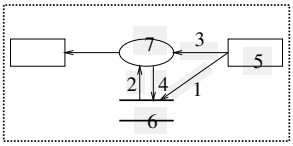
ということが中心となっている．この部分は会議開始前の図にはない部分であるため，ペア 1 の例とは異なり，事前に準備された図の影響は少ないと思われる．図 9, 10, 11からは，それぞれどちらかの作業者が一方的に操作しているので，不一致の危険があると判定できる．図 10では被験者 2 は，自分の最終成果物 (図 8の DFD2) に相当する部分をなぞっている．図 12の区間の始めは，他の図 (STD) に関する区間と重複している．この区間において，被験者 1 が図 8の DFD1' に相当する要素を追加するが，それが追加された後も，被験者 2 は図 8の DFD2 の可能性をなぞっており，被験者 1 による要素の追加が追認されていないことがわかる．よって，この部分の認識の一致は危険であると予想できる．

図 4. ペア 1 の重大な不一致



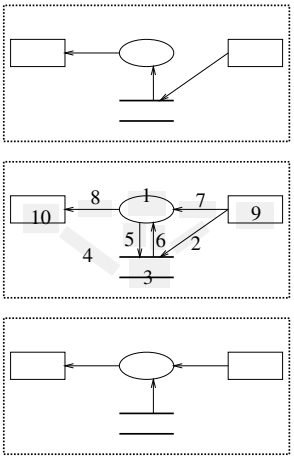
操作対象	操作者	操作の種類
1	被験者 1	point
1 2	被験者 2	draw
2 3 1 4	被験者 2	trace
2 3 5 6 7 4	被験者 1	trace
1	被験者 2	delete
6 3	被験者 2	point
3 5 6	被験者 2	trace
2 5	被験者 2	draw
2 3 5 6	被験者 2	trace
6	被験者 2	point
6	被験者 1	point
3 5 6 7 4	被験者 1	trace
8	被験者 1	point 否定
2 3 5 6 7	被験者 1	trace

図 5. 図 4の不一致に関する区間 (64 分から 66 分) の図の変遷と操作列



1 2	被験者 1	trace
3	被験者 1	point 否定
3 4	被験者 2	delete
7	被験者 2	point

図 6. 図 4の不一致に関する区間 (70 分から 71 分) の図の変遷と操作列



1	被験者 2	point
2	被験者 1	delete
3 4 10	被験者 1	point
5 6 7	被験者 1	draw
1 5 3	被験者 1	trace
3 6 1 8 10	被験者 1	trace
8	被験者 1	point
1	被験者 1	point
1 5 3	被験者 1	trace
1 8 10	被験者 1	trace
8	被験者 2	point
3	被験者 1	point
5	被験者 1	delete
7	被験者 1	delete draw

図 7. 図 4の不一致に関する区間 (72 分から 78 分) の図の変遷と操作列

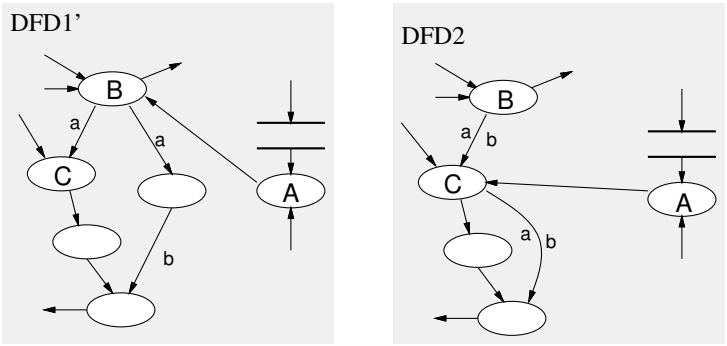
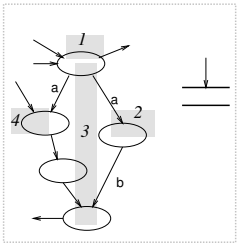
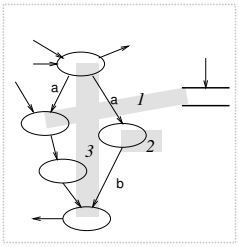


図 8. ペア 2 の重大な不一致



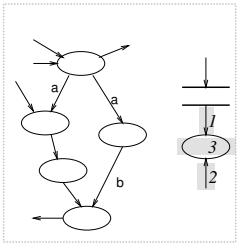
操作対象	操作者	操作の種類
1 2	被験者 1	point
3	被験者 1	trace
4	被験者 1	point

図 9. 図 8の不一致に関する区間 (26 分から 30 分) の図の変遷と操作列



1	被験者 2	trace
2	被験者 2	point
3	被験者 2	trace

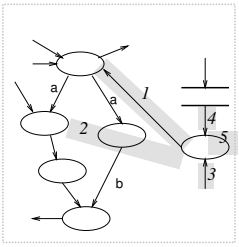
図 10. 図 8の不一致に関する区間 (37 分から 38 分) の図の変遷と操作列



(39-40)

1 2 3	被験者 1	draw

図 11. 図 8の不一致に関する区間 (39 分から 40 分) の図の変遷と操作列



1	被験者 1	draw
	STD の方の区間が入る	
1	被験者 2	trace
2	被験者 2	trace
3 4 5 1	被験者 1	trace

図 12. 図 8の不一致に関する区間 (42 分から 43 分) の図の変遷と操作列

5 おわりに

本稿で提案した方法は，一部は有効であるが，イニシアティブの決定や，区間開始者，終了者の扱いは再考する必要がある．また，図の切り替わりを利用して時区間を限定することで，作業を単純化しているが，近接する区間と，その区間で議論されている図に関する情報を利用することも必要であると考えられる．

図に対する操作と操作者の認識は現在の計算機でも可能であると思われるが，視線や発話によるポイント，他者の指示による図に対する操作などがある場合，本方法の適用は困難となる．この場合，支援システムのオペレータなどが，これらの情報を計算機に対して与える必要がある．また，本手法では，話者交替の情報は利用していないが，その情報を用いて，より明確な検出をすることが可能であると思われる．話者交替の情報を利用した話題の認識は文献 [6] におけるシステムに実装されている．さらに，単純に認識できる範囲の操作や発話の言語行為情報を用いれば，より強力な検出を期待できるが，それらの認識を計算機で行なうことは困難であるため，前述の視線などの情報と同様に人間による補間作業が必要となる．

OMT のような作成要素 (データフロー, データストアなど) の認識が容易な作業では，本稿の手法は容易に適用できるが，構成要素の認識が難しい議論では本方法の適用が困難となる．しかし，最終的に図や文書などの生産物を作成することを目的とする共同作業の場合，その生産物の構造は，ある程度予測ができるため，そのような目的の作業に適用する限りにおいて，問題にならないと思われる．

謝辞

本研究の一部は，文部省科学研究費 奨励 (A) 課題番号 08780257 の援助の下に実施された．記して謝意を表する．

参考文献

[1] B.Boehm. Industrial software metrics top ten list. *IEEE Software*, Sep. 1987.

[2] K. Hori. A System for Aiding Creative Concept Formation. *IEEE Trans. Systems, Man, and Cybernetics*, Vol. 24, No. 6, pp. 882–894, 1994.

[3] 女部田武史. 複数の KJ 法図解の差異や共通部を可視化するシステムの実現について. 第 18 回システム工学部会研究会「発想支援技術」資料, pp. 21–28, 1995.

[4] 桑名栄二. ソフトウェア履歴利用の研究動向. 電子情報通信学会誌, Vol. 77, No. 5, pp. 531–538, May 1994.

[5] T. Uetake, M.Nagata, K.Takeuchi, and H.Takagi. A meeting support system based on analyses of human-communication - using the context and contents of meetings -. *Proceedings of HCI93, Volume2*, pp. 967–972. Elsevier, Aug. 1993.

[6] Haruhiko Kaiya, Motoshi Saeki, and Koichiro Ochimizu. Design of a Hyper Media Tool to support Requirements Elicitation Meetings. CASE'95 Proceedings, pp. 250–259, IEEE Computer Society Press.

[7] James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorensen. *Object-Oriented Modeling and Design*. Prentice-Hall, 1991.

[8] 山崎利治. 共通問題によるプログラム設計技法解説. 情報処理, Vol. 25, No. 9, p. 934, Sep. 1984.