

# 卒業研究報告書

---

## 題 目

言語・非言語アウェアな  
協調学習プラットフォームの構築

Verbal and Non-verbal Aware Platform for CSCL Systems

---

指導教員 瀬田 和久 林 佑樹

---

平成 27 年（ 2015 年 ）度 卒業

(No. 1121100140 ) 杉本 葵

---

大阪府立大学 現代システム科学域 知識情報システム学類

---

## 目次

1 序論 .....	1
1.1 研究背景 .....	1
1.2 研究目的 .....	4
1.3 本論文の構成 .....	5
2 アプローチ .....	6
2.1 学習支援に向けた学習ツールの拡張 .....	6
2.2 言語・非言語情報の利用 .....	7
2.3 プラットフォームの概要 .....	9
3 協調学習プラットフォーム .....	10
3.1 プラットフォーム構成 .....	10
3.2 クライアント・サーバ間の通信プロトコル .....	11
3.3 参加者認証管理 .....	13
3.4 言語・非言語情報の取得 .....	16
3.5 学習ツールの開発方法 .....	18
4 プラットフォームに基づく協調学習環境 .....	20
4.1 システム利用 .....	20
4.2 実装した学習ツール .....	21
4.3 学習ツールを活用した協調学習支援 .....	26
5 結論 .....	27
5.1 本研究で得られた成果 .....	27
5.2 今後の課題 .....	27
謝辞 .....	29
参考文献 .....	30

## 図目次

図 1.1.1	協調の階層構造 .....	2
図 2.2.1	インタラクションの階層的解釈モデル（角ら <sup>[20]</sup> より引用） .....	8
図 2.3.1	協調学習プラットフォームの概要 .....	9
図 3.1.1	協調学習プラットフォーム構成図 .....	10
図 3.2.1	メッセージ構造 .....	12
図 3.2.2	通信プロトコル階層 .....	12
図 3.2.3	オブジェクトと JSON 形式の変換 .....	13
図 3.3.1	ログイン処理の流れ .....	14
図 3.3.2	ログアウト処理の流れ .....	15
図 3.4.1	言語情報取得処理 .....	17
図 3.4.2	筆記情報取得処理 .....	17
図 3.4.3	筆記情報取得処理 .....	18
図 4.1.1	ログイン画面 .....	20
図 4.1.2	メインウィンドウ .....	20
図 4.1.3	ログイン後のシステムインタフェース .....	21
図 4.2.1	ビデオチャット .....	22
図 4.2.2	テキストチャット .....	22
図 4.2.3	チャットメッセージ .....	22
図 4.2.4	トークメッセージ .....	23
図 4.2.5	トーク履歴 .....	23
図 4.2.6	Web ブラウザ .....	24
図 4.2.7	共有ボード .....	25
図 4.2.8	共有ボードのメッセージ .....	25

## 表目次

表 2.1.1	学習形態の類型化と使用リソース（大山・田口 <sup>[19]</sup> より作成） .....	7
---------	---	---

---

# 1 序論

## 1.1 研究背景

現在、教育現場では教師から学習者への一方向型の授業が多く行われているが、教師と学習者、あるいは学習者同士が意見を交換し、物事を多角的に考察する双方向型の授業も取り入れるべきだといわれており<sup>[1]</sup>、今後、協調学習の導入が考えられる。協調学習では、学習のプロセスや参加者間のインタラクションが重要とされており、参加者全員が主体的に学習を進め、理解しながら議論を展開していくことが望ましい。このため、他者を観察し理解状況を推測したり、議論の状況を俯瞰的に捉え整理したりするスキルを要する。しかし、こうしたスキルは形式化されているものではなく、スキル獲得には経験を要するため、学習者だけで意義のある協調学習を展開することは難しい。議論の展開についていけない学習者がいることや、議論内容が設定されたテーマから大幅に外れてしまうことも少なくない。そこで、教師が介入し、助言することがこうした問題に対する解決策の1つとして挙げられるが、教師一人に対しての学習者数は非常に多く、教師に負担がかかるのは明らかである。こうした背景から、コンピュータで協調学習を支援しようとする CSCL (Computer Supported Collaborative Learning) の研究が注目されている。CSCL は、参加者同士の関わりに計算機が介入し協調学習の支援を目指す研究領域である。情報通信技術が発達し、ネットワーク上でメッセージを交換したり、資料を共有したりすることが簡単にできるようになった現在、CSCL はネットワークを介して様々な人との交流を促し、教育空間を広げるものとして注目されている<sup>[2]</sup>。CSCL と類似した概念としてグループウェアが挙げられる。ネットワーク上で行うグループ活動を支援するという点では CSCL と共通した試みであるが、グループウェアはユーザに共有環境へのインタフェースを提供するシステムのことを示し<sup>[3]</sup>、CSCL はグループ活動の中でも協調的活動の支援を目的としており、そのプロセスや参加者間の相互作用が重要とされる。

---

複数の人間によって行われる協調活動は複雑な要素から構成されるとい

われ、石井<sup>[4]</sup>は協調活動の概念を図 1.1.1 のように示している。下位概念から Awareness（お互いに気づく）、Communication（情報を交換する）、Collaboration（新しい価値を創造する）と上位概念へとつながるこの階層は Collaboration には Communication が必要不可欠であり、また Communication には Awareness が必須であるということを示している。



図 1.1.1 協調の階層構造

誰がいるか見えて、話が聞こえて、何をしているのかがわかる Awareness を支援する研究は、主にグループウェアにおける研究領域で行われてきた。Eisert<sup>[5]</sup>はネットワーク上での視線一致を実現した。複数のカメラで参加者を多角的に撮影することで 3D モデルを作成し、三次元仮想空間に配置する。そして、各参加者の目線に応じた角度からの映像を通信することでネットワークを介した視線一致を可能にした。また、各参加者が何を見ているのかという共同注視を反映した研究もある。Vertegaal<sup>[6]</sup>は赤外線 LED を利用し、仮想空間上の共有資料に各参加者の視線箇所を表示できるようにした。これにより、分散環境上の参加者は他の参加者が何を見ているのか認識できるようになった。岡田・松下<sup>[7]</sup>は、曲面上のスクリーンに等身大のビデオ映像を投影し、これまで断片的だった画面を連続させることで臨場感あるテレビ会議システム MAJIC を開発した。

このように、グループウェアでは遠隔地にいても互いの非言語的な振る舞いがわかるための様々な研究が進められてきた。一方で、質の高い Collaboration を目指す CSCL の研究は大きく分けて 2 種類あり、協調学習のプロセスには立ち入らずに、グループの形成や、学習目的、環境を設定することで間接的に支援するものと、協調学習のプロセスに直接関与するものがある。

---

前者に該当する研究として、グループ構成員を提案することで議論の活性化を支援する研究がある。今井ら<sup>[8]</sup>は参加形態がオープンな開放型グループ学習環境において、ネットワーク上で学習者の仲介を行う Mediator Agent (以下, MA) を提案している。学習者が何らかの疑問を持って討論要求を MA に依頼する。MA は他の学習者情報を参照し、その疑問に回答可能と思われる学習者や、その話題に興味がある学習者を選出し討論の提案を行うというものである。また、“Six thinking hats”<sup>[9]</sup>を適応した協調学習支援システムがある<sup>[10][11]</sup>。“Six thinking hats”では、参加者が被る帽子の色に合わせて議論を行っていく。例えば白い帽子であれば、事実に基づいたデータを元にして意見を主張する一方、赤い帽子であれば思ったままの意見を述べるなど与えられた役割に応じて思考していくことで、一つの処理を集中して実行でき、グループとして多角的な意見が得られる手法として有効だといわれている。

協調学習のプロセスに関与する研究では、中村ら<sup>[12]</sup>の擬似学習者を導入する方法がある。学習課題に対する領域固有知識を持つコンピュータエージェント（擬似学習者）を参加者の 1 人として導入することで、参加者全員に共通して欠落している知識の補完を促し、議論を円滑に進めることを試みた。この擬似学習者を導入する手法はコンピュータが学習課題に関する領域固有の知識を所持すること、及び参加者は互いの顔が確認できない環境で協調学習を行うことが前提となっている。一方で、学習課題内容に立ち入らず、協調学習を支援しようとする研究もある。稲葉ら<sup>[13]</sup>は、参加者が発言する際、発言内容だけでなく、カテゴリ化した意図を選択させることで、学習のインタラクションプロセスを抽象化して可視化するシステムを開発した。また、Baker & Lund<sup>[14]</sup>が提案した C-CHENE システムは参加者が発言する際、発言内容の文頭語句を選択して発言させることで、内容に関連した議論を展開させた。これらはテキストチャットを用いた協調学習支援システムとなっており、言語情報を扱っている。

---

## 1.2 研究目的

このように、様々な CSCL の研究が行われてきたが、ここで、2 つの問題が挙げられる。

1 つは様々な学習状況に対応可能な基盤が必要ということである。既存の協調学習支援システムは、それぞれが特定の学習状況を前提として開発されてきた。テキストチャットを用いて非同期で行うものもあれば、ビデオチャットを利用した同期型の学習では、即時的なインタラクションで議論が展開される場合もある。また、学習課題やグループの構成員によっても学習中にやりとりされるインタラクションの種類は異なる。共有のボードを利用して教えあう状況もあれば、インターネットを利用して調べ学習を行う状況、プレゼンをして他の参加者が質問や意見を述べる状況など様々な状況が考えられる。このような特定の状況を前提として独自にシステム開発が行われてきた。協調学習を支援するという目的が共通しているにもかかわらず、独自に研究がされてきたことにより共通する機能を共有できず、開発されたシステムの発展・実用化が困難となっているのが現状である。

2 つ目の問題は、言語・非言語情報を扱えていないことである。対話は言語情報のみを用いて行っているわけではない。グループウェアの研究領域では視線一致や共同注視ができたり、他者の身振りがわかったりなど、ネットワーク環境上でも相手の非言語的な行動が見えるための研究が進められてきた。対話場面において非言語情報は重要な要素であるにもかかわらず、CSCL の研究領域において、計算機が参加者の非言語的振る舞いを利用して支援する研究はそう多く存在しない。参加者の非言語的情報を扱った研究として、同一空間内で行われる協調学習の学習状況を可視化する試みが行われてきたが<sup>[15]</sup>、学習後に分析するという段階に留まっており、こうした知見を取り入れ、リアルタイムにフィードバックを与えることが望まれる。

そこで、本研究ではあらゆる学習状況に対応可能で言語・非言語情報の扱いを操作できる協調学習プラットフォームの構築を行う。本プラットフォームに基づき、言語・非言語情報を扱える協調学習支援システムの開発を可能とする。プラットフォームに基づくシステム開発が実現することで、シス

---

テム開発者が想定する学習状況に応じて必要なものを組み込み、取得された参加者の言語・非言語情報を扱い、協調学習を支援するためのフィードバックを与えることが期待される。

### 1.3 本論文の構成

第2章では、あらゆる学習状況に対応可能で、言語・非言語情報を高次に解釈するためのプラットフォームの要件を挙げる。

第3章では、本研究で構築したプラットフォームについて述べる。システム開発者がプラットフォームに基づき拡張していく際に、プラットフォーム内ではさまざまなメッセージをやりとりする。本プラットフォームで実装した任意のメッセージを通信するための通信プロトコルについて述べる。また、取得可能とした言語・非言語情報とその取得方法を説明し、システム開発者がプラットフォームに基づきシステム開発を行う際、どのように拡張していくのか述べる。

第4章では、システム開発者として実装したシステム仕様と学習ツールについて述べ、第3章で述べた通信プロトコル、及び言語・非言語情報を用いてどのようなメッセージをやり取りしたのか、言語・非言語情報をどのように表示したのか説明する。

第5章では本研究での成果をまとめ、協調学習プラットフォームの今後の課題、及び発展性について述べる。



---

## 2 アプローチ

### 2.1 学習支援に向けた学習ツールの拡張

協調学習とは、教師と学習者、あるいは学習者同士が意見を交換したり、課題を解決したりする学習である。協調学習では学習課題に対する成果と共に学習のプロセスや参加者同士の相互作用が重要とされる<sup>[3]</sup>。以下では稲葉・豊田<sup>[3]</sup>が紹介する協調学習の有効性を主張する学習理論をいくつか取り上げる。

#### 1. Peer Tutoring<sup>[16]</sup>

学習者が相互に助け合い、教えあうことによって学習する教授体系である。教わる側の知識獲得、動機付けだけでなく、教える側の学習者にも自己の知識を外化することで知識の強化、理解の深化が期待される。

#### 2. Observational Learning<sup>[17]</sup>

他者の問題解決行動や振る舞いの観察を通して間接的な知識獲得をねらいつける。後に類似した状況に遭遇した際、観察によって得られた知識が適用される。

#### 3. Self-regulated Learning<sup>[18]</sup>

他者との意見交換を通して自己の知識状態、思考過程を認識する機会が生じるとされ、自己の思考をモニタリングし、診断・評価を行い、後の行動を制御する高次の機能の向上につながる。

上記は数例に過ぎないものであり、協調学習には様々な学習効果がある。また、これらの学習理論からもわかるように、協調学習の構成員には教える人、教わる人、観察する人など各役割があること、及び、知識獲得、理解深化など役割に応じた学習目的もあれば、思考モニタリングスキル向上など参加者全員に共通する目的もある。

このように協調学習では役割、目的に応じて様々な学習状況が想定される。大山・田口<sup>[19]</sup>は、大学でグループ学習を導入した授業実践例を整理し、学習の事前作業の有無、及び事後作業の形式から学習形態を表 2.1.1 のように類型化した。そこで各学習形態に応じて使用リソースが異なると述べている。例えば、事前作業がなく事後作業はグループで行う課題解決型の学習では、調べ学習を行う状況があるため、図書館、インターネットなどを利用することがある。また、事前作業があり事後作業は個人で行う理解深化型の学習では、学習を通して他者から得た意見によって事前作業から事後作業までの活動を通してどのように自己の思考が変化したのか可視化するためのワークシートを利用することがあるとしている。複数の参加者が協調的な活動を行う場で使用するツールは様々なものが上げられるため、協調学習を支援する際、システム開発者が参加者に提供する学習ツールは異なるといえる。そこで、本プラットフォームに基づきシステム開発者が学習ツールを組み入れることがプラットフォームの要件として挙げられる。

	事前作業	事後作業	目的	使用リソース
1. 交流型	なし	なし	他者の意見を知り、 他者を理解する	アイスブレイクのためのツール
2. 意見獲得型	なし	個人	他者の意見を傾聴し、 自分の意見を明確にする	議論の軸を提示するワークシート
3. 課題解決型	なし	グループ	課題に取り組み、 成果物を作成する	図書館やインターネット
4. 主張交換型	あり	なし	自己の主張を伝え、 他者からの意見を求める	Web上での議論環境
5. 理解深化型	あり	個人	他者の意見を聞き、 考えを再構築して理解を深める	思考の変化が可視化されるワークシート
6. 集約型	あり	グループ	予めまとめたものを1つに集約し、 作品として表出する	意見を合意形成するためのワークシート

表 2.1.1 学習形態の類型化と使用リソース（大山・田口<sup>[19]</sup>より作成）

## 2.2 言語・非言語情報の利用

協調学習においてやりとりされるインタラクションを支援するためには、“誰が参加できていない”“話題が本来議論すべきことから外れて発散している”といった人間が理解できる高次の状況を計算機が捉えることが必要で

ある．角ら<sup>[20]</sup>は多人数会話場面において，計算機が言語・非言語的データを元に高次の解釈を実現するために，図 2.2.1 のようなインタラクションの階層的解釈モデルを提案している．これは言語・非言語的振る舞いから座標や波形などのデータを取得する Raw Data 層，“誰を見ている”“発言している”という単一の動作，振る舞いへと抽出する Interaction Primitive 層，各参加者から得られた要素を組み合わせる“誰と誰が何を見ている”“誰と誰が互いを見て話をしている”といった現象を検出する Interaction Event 層，時空間的要素を組み入れ“発言権が誰から誰に転換した”“誰が会話に参加した”といった会話状況の流れを理解する Interaction Context 層へと段階的な解釈の積み上げ構造を示している．本プラットフォームでは，言語・非言語データを取得し，単一の振る舞いへと抽出することで，システム開発者がプラットフォームから得られる情報を基に，検出したい状況に応じて各要素を統合し高次に解釈できるようにする．

また，このような段階的に解釈するモデルがあるものの，実用的解釈ルールがそう多く存在しないのが現状である．解釈ルールを作り上げるためには実データを収集し，分析に基づいて仮説を立て，検証するというプロセスを

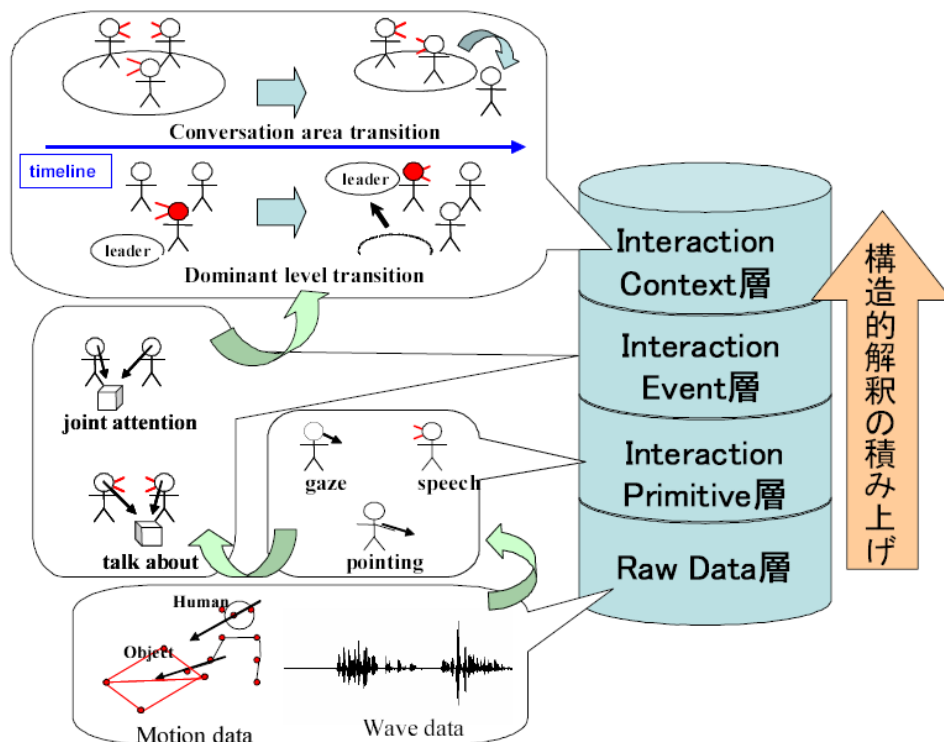


図 2.2.1 インタラクションの階層的解釈モデル（角ら<sup>[20]</sup>より引用）

繰り返す必要がある。しかし、これまで行われてきた分析手法は、ビデオ映像に基づいて人が分析することや、デバイスを用いてデータを取得するにしても、大規模なセッティングし、様々な機器を人に装着して行うものであり、データの取得が終わってから分析を行うものであった。このように人的コストと時間を要し、分析に負荷がかかりすぎるため、検証に至るまでのプロセスを繰り返すことが困難であった。本プラットフォームでは容易にセッティングできる言語・非言語情報取得デバイスを使用し、リアルタイムでデータを解析できる環境を整える。

### 2.3 プラットフォームの概要

図 2.3.1 に協調学習プラットフォームの概要を示す。システム開発者は独自の学習ツールを本プラットフォームに組み入れ、学習状況に応じた学習ツールをユーザに提供する。また、言語・非言語情報を操作できるので、組み合わせで解釈につなげることや、学習ツールに反映して参加者に見せることができる。プラットフォームの内部でやりとりされる処理を意識せずに、拡張することができるため、容易に言語・非言語情報を扱えるシステム開発が可能となる。

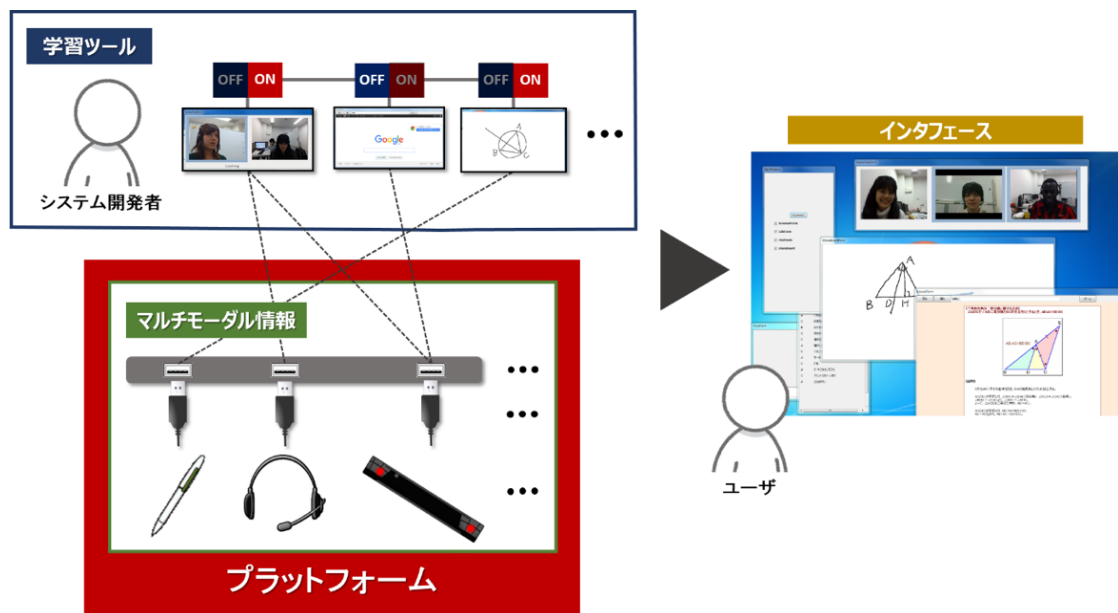


図 2.3.1 協調学習プラットフォームの概要

### 3 協調学習プラットフォーム

#### 3.1 プラットフォーム構成

協調学習プラットフォームの構成は図 3.1.1 に示すようなクライアント・サーバ型の構成となっている。サーバは、ビデオ映像をストリームで通信するサーバ（以下、ビデオ映像通信サーバ）とテキストで参加者リストの管理や学習ツールを同期させるための情報、言語・非言語情報などを通信するサーバ（以下、協調学習管理サーバ）がある。両サーバ共に、Java 言語で実装し、ストリームサーバは RTMP プロトコルに準拠したオープンソースの Red5<sup>[21]</sup>を利用した。クライアントは C#で実装した Windows アプリケーションとなっている。なお、本プラットフォームに基づいたシステムを実行する場合、視線取得デバイスである Tobii 社が製作したアイトラッカの EyeX ドライバ<sup>[22]</sup>、及び Adobe の Flash Player<sup>[23]</sup>のインストールが必要である。

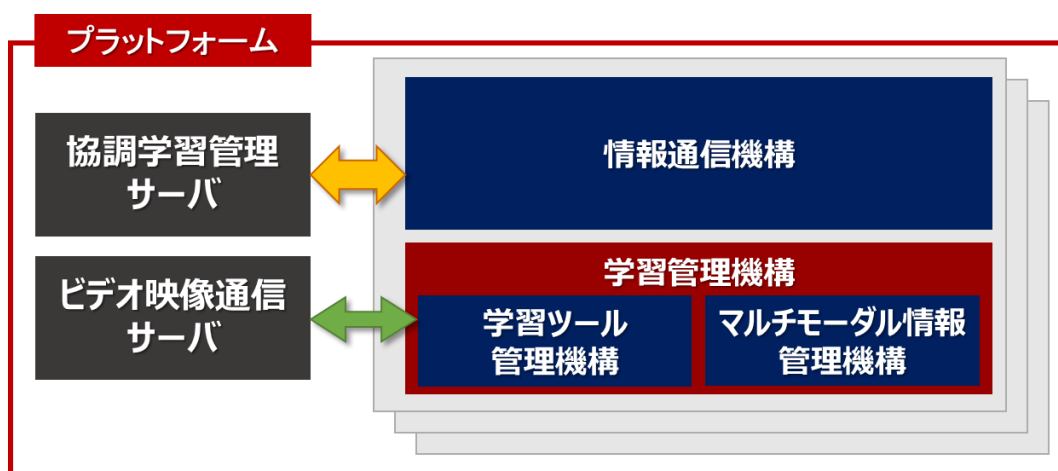


図 3.1.1 協調学習プラットフォーム構成図

クライアントは、学習管理機構、及び情報通信機構から構成され、学習管理機構は学習ツール管理機構、マルチモーダル情報管理機構から成る。学習ツール管理機構ではシステム開発者が実装した学習ツールを定義し、参加者に提供する学習ツールを選択する。マルチモーダル情報管理機構では本プラットフォームで取得可能な言語・非言語情報が定義されている。情報通信機構には協調学習管理サーバと任意のメッセージを通信する機能が備わって

---

いる。任意のメッセージを交換できることで、取得する言語・非言語情報を追加する際にクライアント・サーバ間で行われる通信処理の流れを考慮せずにプラットフォームを拡張できる。また、システム開発者もクライアント・サーバ間での通信処理を意識することなく任意のメッセージを通信できることで、より簡単に学習ツールを組み入れることができる。任意のメッセージをやり取りする通信プロトコルについては 3.2 で述べる。

ストリームサーバでは、ビデオチャットと映像データ、音声データの通信を行う。映像データや音声データの通信はネットワークの状況によっては負荷がかかり、遅延が生じる問題が考えられるため、ネットワークの帯域幅に合わせて映像の質をシステムで設定している。ビデオチャットは **Flash** で動作するアプリケーションを製作し、フォーム内に **Shockwave Flash** オブジェクトとして埋め込むことで実装した。**Flash** アプリケーションとビデオ映像通信サーバ間でやり取りされる通信は、ログインした際、既に参加している人の映像を正しく取得したり、新たに参加する人がいればその新規参加者の映像を取得したりするなど、参加者のログイン・ログアウトに応じた処理が必要になる。協調学習サーバが管理する参加者のログイン状況に応じてビデオチャットも正しく映像を通信しなければならない。3.3 ではログイン・ログアウトの際の処理の流れを説明する。

### 3.2 クライアント・サーバ間の通信プロトコル

クライアント・サーバ間で任意のメッセージを交換するプロトコルについて図 3.2.1 のようにシステム開発者が考慮しなければならない開発者層と実際に通信をやりとりするプラットフォーム層に分けて説明する。

開発者層では、メッセージをオブジェクトで通信する。メッセージのオブジェクトは図 3.2.2 のような構造をとっている。全メッセージクラスを抽象化した **Message** クラスを用意し、メッセージの種類を表す **msgtype** プロパティを付与する。システム開発者はこの **Message** クラスを継承したクラス

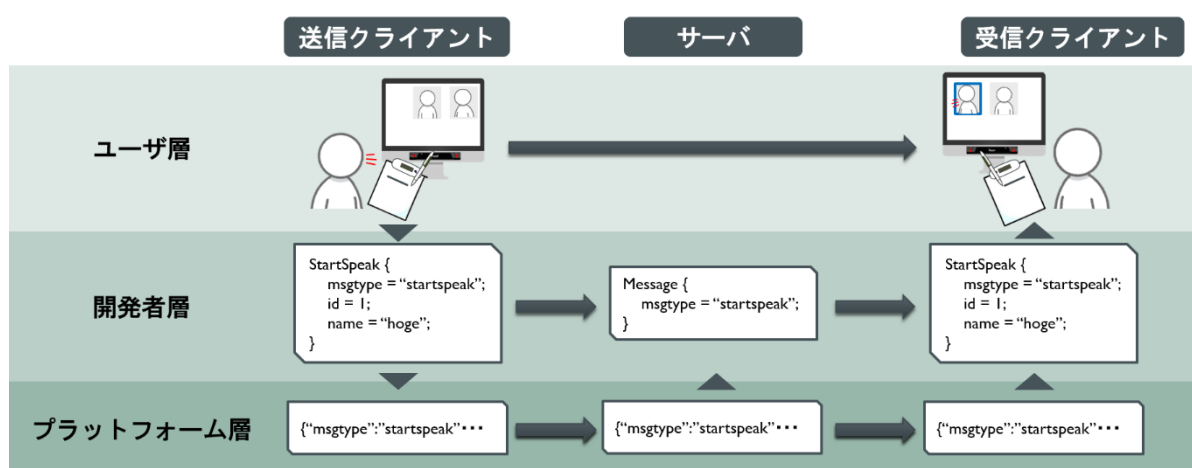


図 3.2.2 通信プロトコル階層

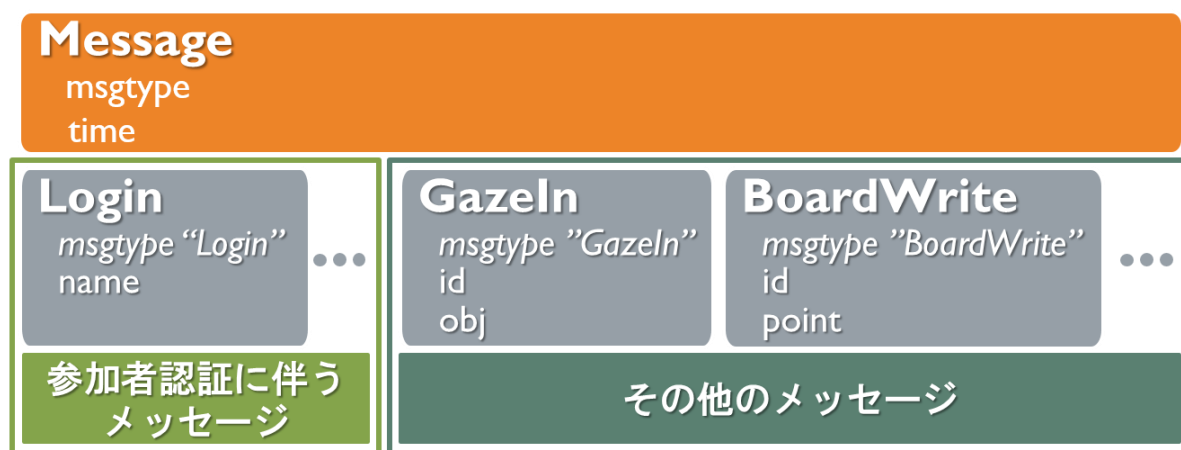


図 3.2.1 メッセージ構造

を定義する。定義されたクラスの `msgtype` プロパティに固有の値を代入することで、その他任意のプロパティを含む独自のメッセージクラスを作ることができる。

ここで、`Message` クラスを継承するクラスには大きく分けて2種類ある。1つは `Login`, `Logout` など参加者管理に伴うメッセージである。参加者情報管理はサーバが行っているため、この種類のメッセージはクライアント・サーバ共に定義されている。また、システムに共通して行われる処理に伴うメッセージであるため、既にプラットフォーム上で定義されている。一方、その他のメッセージは種類に限りがない。`Message` クラスを継承し、`msgtype` プロパティにメッセージ固有の値を代入すればシステム開発者が独自に定義できるものである。この種類のメッセージはクライアントのみで定義され、サーバではタイプを参照しログイン・ログアウトに関係しないメッセージで

あればデータをそのまま全クライアントに送信するため、システム開発者はクライアント側でのみメッセージを定義すれば任意のメッセージをやりとりすることができる。

このように独自に定義されたメッセージオブジェクト全てをプラットフォーム層では JSON 形式に変換して通信を行っている。図 3.2.3 にオブジェクトと JSON 形式のメッセージ交換例を示す。オブジェクトと JSON 形式の変換はプロパティの“変数名”：“値”の集合として表現される。プラットフォームではオブジェクトを JSON 形式に変換する機能を用意したため、システム開発者はこの JSON 形式への変換を意識せずにサーバとメッセージを交換することができる。

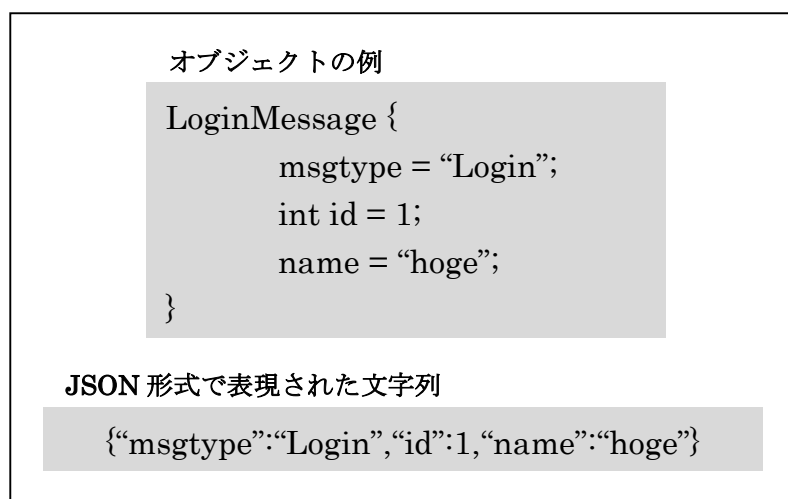


図 3.2.3 オブジェクトと JSON 形式の変換

### 3.3 参加者認証管理

図 3.3.1, 及び図 3.3.2 は参加者がログアウト・ログインした際に起こるクライアント、協調学習管理サーバ、ビデオ映像通信サーバの処理の流れを示す。ログイン処理では、新規参加者が自分の映像をビデオ映像通信サーバに送信してから、既存参加者が新規参加者の映像を受信する流れになっている。ログアウト処理は、既存参加者全員が退出者の映像を受信停止完了してから退出者がログアウトする流れとなっている。



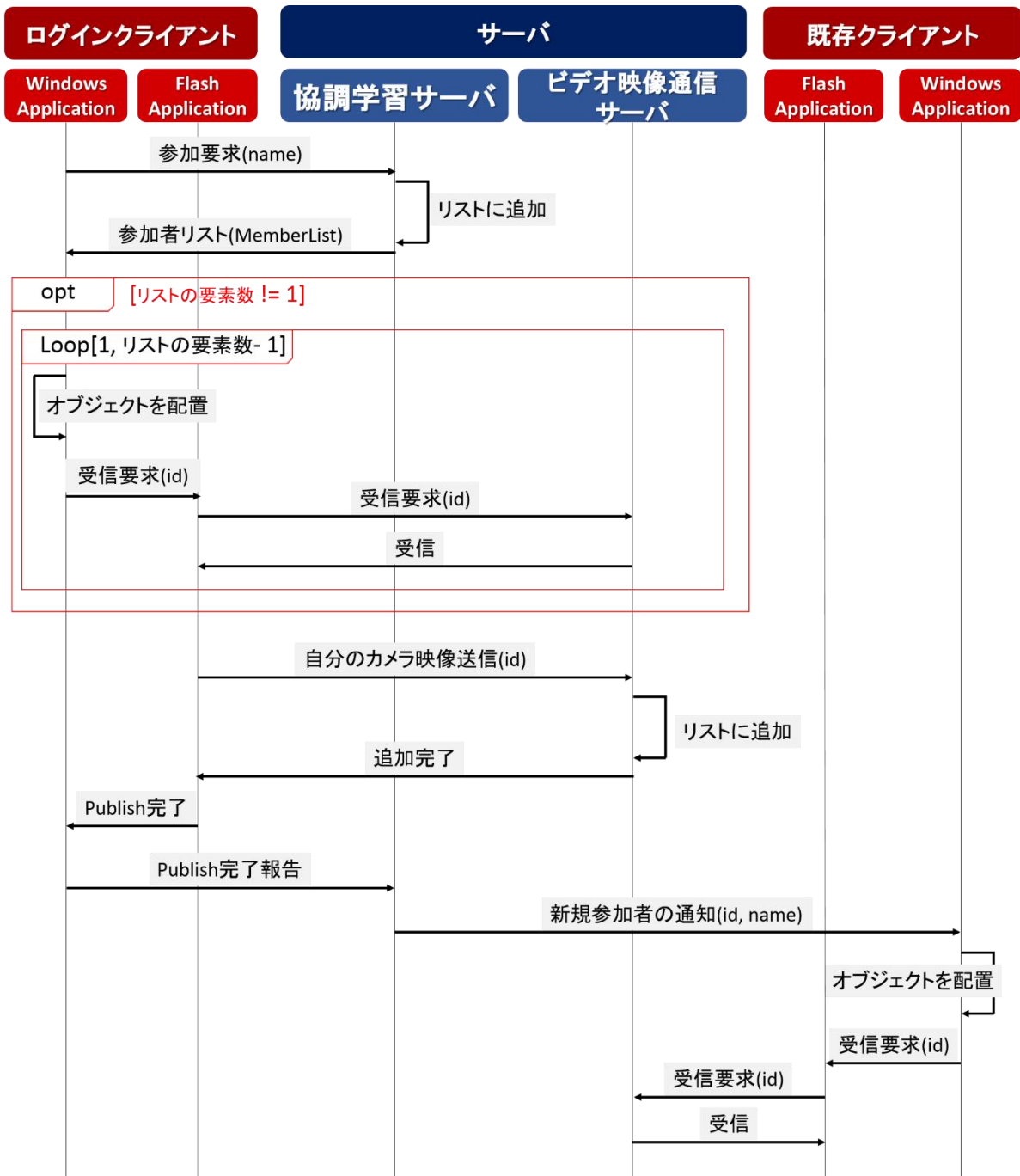


図 3.3.1 ログイン処理の流れ

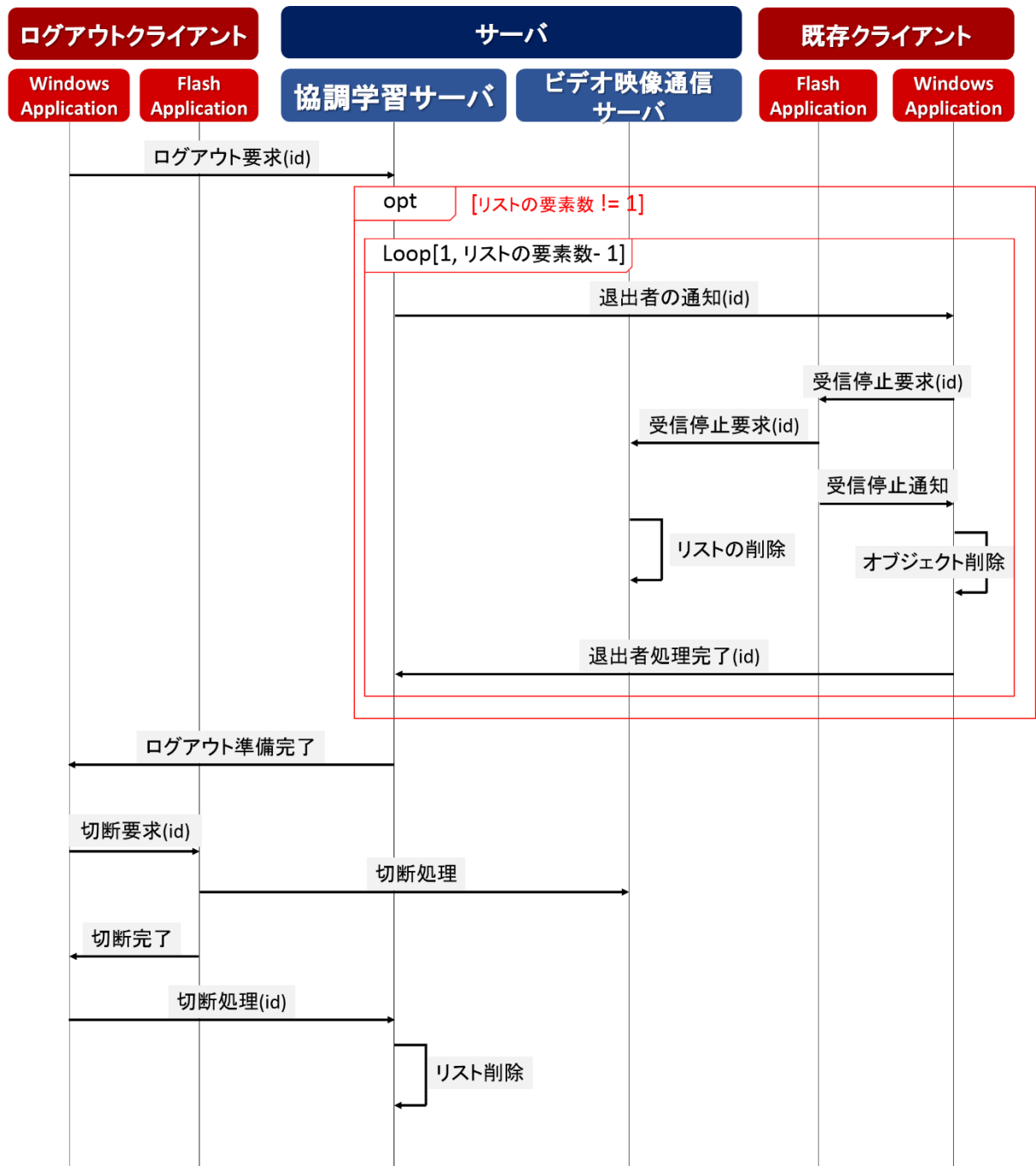


図 3.3.2 ログアウト処理の流れ

---

### 3.4 言語・非言語情報の取得

本プラットフォームでは発話情報, 筆記情報, 視線情報を取得可能にした。これらの情報は協調学習中に行われるインタラクションの要素として特に重要といえるからである。適切に議論を可視化することは、議論が活性化することにつながるといわれており<sup>[24]</sup>, 発話情報が重要なものであるのは明らかである。筆記情報を利用して発話者に貢献への気づきを与え参加意識が向上することが確認されており<sup>[25]</sup>, 筆記情報も重要な要素といえる。また、議論が望ましい状況と望ましくない状況で展開されている事例を分析し、違いの 1 つとして参加者の視線を挙げている研究<sup>[26]</sup>があるように、視線情報は協調学習の状況を認識する上で重要な要素となると考えられる。

うなずきや表情, 身振りなど他の非言語情報も重要な情報であるが、発話情報・筆記情報・視線情報を取得することは特に重要だと考え、本研究で取得可能とした。以下では各情報の取得方法について述べる。

#### 1. 発話情報【発話タイミング・発話内容】

本プラットフォームは、音声認識アプリケーション（以下、Speech Recognition）を起動し Speech Recognition がコンソール出力する発話タイミング、発話内容を読み込むことで言語情報を取得している。図 3.4.1 に言語情報取得の処理を示す。Speech Recognition は音声認識エンジン Julius<sup>[27]</sup>付属の adintool を利用し、マイクからの入力音量が閾値を超えたら発話開始、下回ったら発話終了とみなすことで発話区間をリアルタイムで取得し、音声ファイルを生成する。Google Speech API<sup>[28]</sup>を利用して生成された音声ファイルから発話内容を取得する。

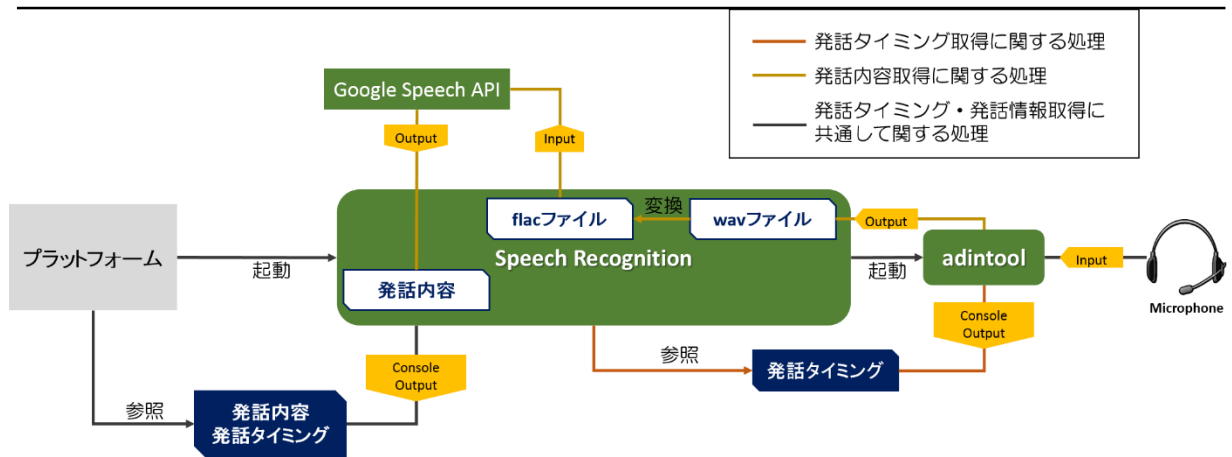


図 3.4.1 言語情報取得処理

## 2. 筆記情報【筆記タイミング】

筆記情報の取得は Pentel 社のデジタルペン<sup>[29]</sup>を利用する。図 3.4.2 に筆記情報を取得する処理の流れを示す。筆記情報はプラットフォームが、筆記タイミングを取得するアプリケーション（以下、Write Recognition）のコンソール出力内容を読み込むことで筆記情報を取得している。Write Recognition では、デジタルペンの筆記状態を常にコンソール出力する。具体的には、1：ペンが紙面につく瞬間、2：ペンが紙面についている、3：ペンが紙面についていない、の3種類がリアルタイムに出力されており、プラットフォームではこの出力内容を常に読み込み、1を筆記開始のタイミング、2から3に出力内容が変わった瞬間を筆記終了のタイミングとして検知している。

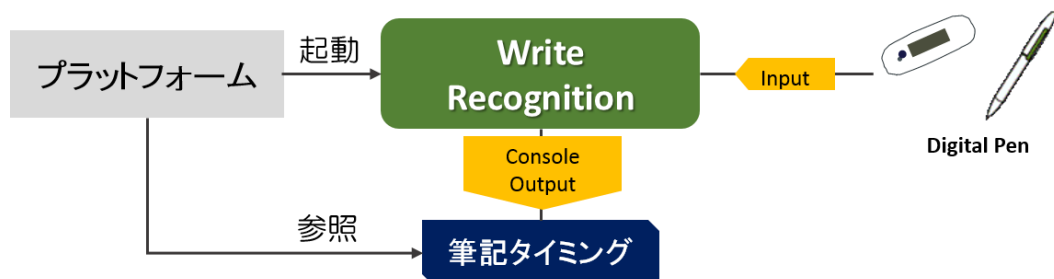


図 3.4.2 筆記情報取得処理

### 3. 視線情報【視線タイミング・視線対象オブジェクト】

視線情報の取得には Tobii 社のアイトラッカを利用する。アイトラッカはディスプレイに装備し、ユーザが事前にキャリブレーションを行うことで、ディスプレイにおけるユーザの視線座標を取得できる。Tobii 社が提供する C#用のライブラリ EyeX SDK を利用し、C#で定義したオブジェクトに関心領域 AOI (Area of Interest) を付与することでオブジェクト上に視線があるかないかを検出できる。ここでは、どの学習ツールを見ているのかという学習ツール単位でのオブジェクトに加え、誰のビデオ映像を見ているのか、誰のどのコメントを見ているのかなど、学習ツール上に定義されている各オブジェクトに AOI を付与することで視線検知できる仕組みを採用している。



図 3.4.3 筆記情報取得処理

### 3.5 学習ツールの開発方法

本プラットフォームに基づいてシステムの開発者は開発を行う際、どのように学習ツールや、言語・非言語情報の解釈機構を組み入れるのかについて説明する。

学習ツールの実装についてはプラットフォームのクライアントと同様に C#で実装する。別の環境で開発したアプリケーションを学習ツールとして実装した場合、外部のアプリケーションとして起動することはできるが、協調学習の分析・支援につなげる際、参加者が学習ツールに対して行った振る舞いの認識や、受信したメッセージに基づいて学習ツール上に何か表示する

---

ことを実現するためには C#で実装することが必須となる．ただし，ビデオチャットのようにフォーム上に外部のアプリケーションを組み入れることができれば，そのアプリケーションを 1 つのオブジェクトとして AOI を設定することができる．また，サーバとやりとりする情報がある場合はメッセージを定義すればプラットフォーム内の情報通信機構の機能を利用してサーバに情報を送ることができる．

学習ツールを追加する際は，プラットフォーム内の学習ツール管理機構を修正して新しい学習ツールを定義する．また，本プラットフォームに基づくシステムは学習ツールをウィンドウ別に実行させるインタフェースとなっているため，レイアウトを作りこむ必要はない．

## 4 プラットフォームに基づく協調学習環境

### 4.1 システム利用

本研究でプラットフォームに基づき開発したシステムの仕様について述べる．図 4.1.1 は本システムのログイン画面を示す．ユーザはサーバアドレスや，ポート番号，名前を入力するほか，複数ディスプレイがある場合，視線取得デバイスであるアイトラッカ検知用のディスプレイを選択する仕様となっている．

ログイン後は図 4.1.2 のような 1 つのメインウィンドウと，各学習ツールウィンドウから構成される図 4.1.3 のような複数ウィンドウのインタフェースになる．メインウィンドウでは使用する学習ツールをチェックボックスで選択する仕様となっており，全ての学習ツールを制御する．そのため，各学習ツールは学習ツールのウィンドウ上で画面を閉じることはできず，メインウィンドウでサーバとのコネクションを解除するとその他学習ツールのウィンドウも全て閉じる仕様になっている．また，全ての学習ツールは抽象クラスによって制御されており，ログインの際に指定したディスプレイの範囲内でのみ位置やサイズを変更することができる．

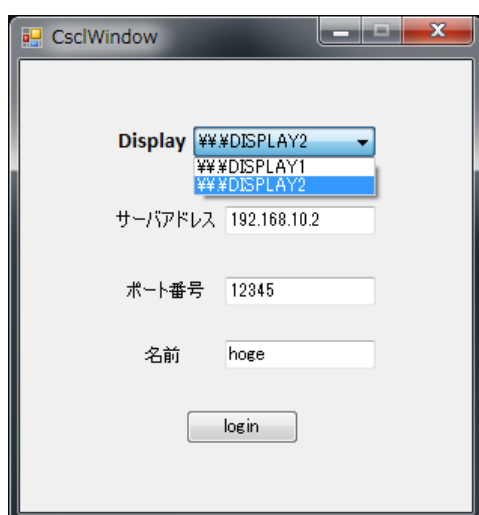


図 4.1.1 ログイン画面

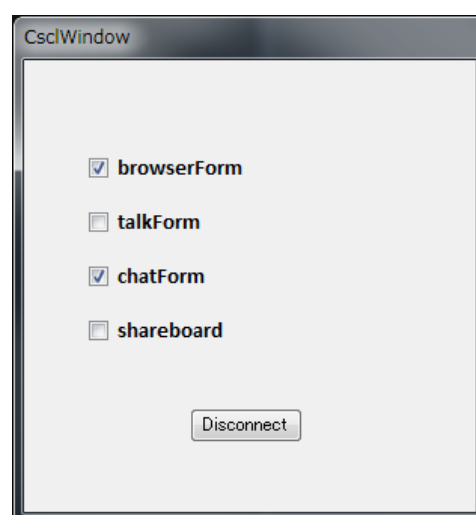


図 4.1.2 メインウィンドウ

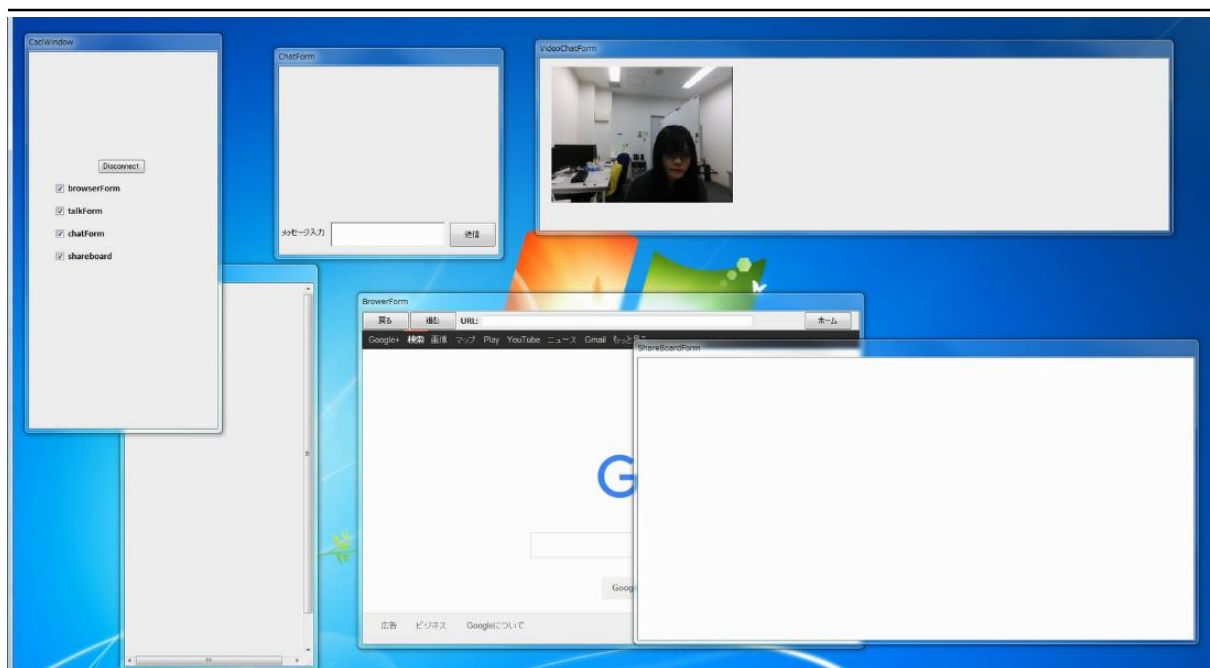


図 4.1.3 ログイン後のシステムインターフェース

## 4.2 実装した学習ツール

本プラットフォームに基づいて実装した学習ツールの仕様, 及びサーバとどのような通信をして学習ツール上で何が実現できるのかを説明する.

### 1. ビデオチャット

ビデオチャットは分散環境における協調学習で相手の顔が見え, 振る舞いが見えるため重要なツールである. コミュニケーションの要素として言語情報はもちろん重要であるが, 非言語情報も非常に重要な役割を果たす. そのため, 本システムではビデオチャットは参加者がメインウィンドウで操作せず, 常に表示する仕様となっている.

ログイン時に参加者のリストを受信することで既存参加者の映像を表示し, 入出者がいるとサーバから随時メッセージを受信することでビデオ映像のオブジェクトが追加・削除される. また, フォーム上では各映像に AOI を定義しているため, 視線がどのオブジェクト上



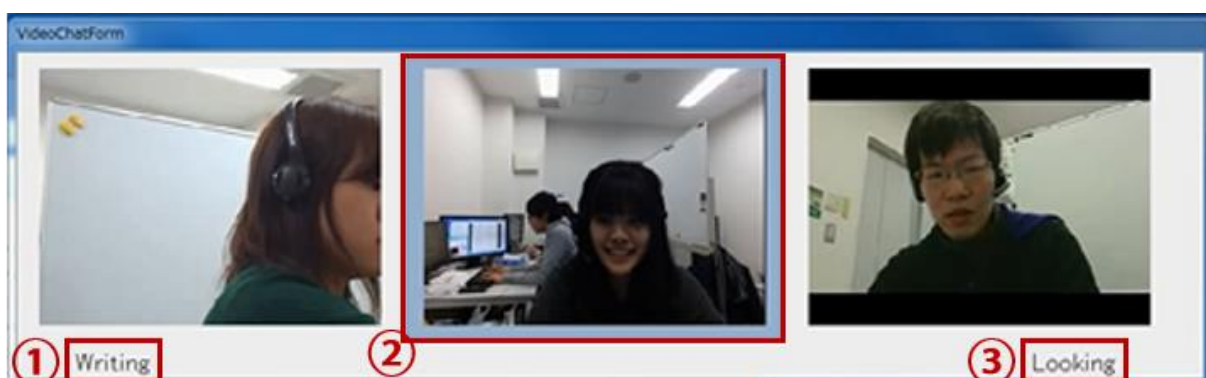


図 4.2.1 ビデオチャット

にあるのか、つまり誰を見ているのかということを確認することができる。このビデオチャットでは「メモを取っている」、「発話している」といった単一の言語・非言語情報を受信し、ツール上に反映することが可能となっており、①のようにメモを取っている人のラベル表示や、②のように発話している人物の映像を強調して表示している。また、視線情報のメッセージを受信した場合、メッセージのプロパティに含まれる視線対象人物の id が自分の id と一致すれば③のようにメッセージ送信者が自分を見ていることをラベル表示する。

## 2. テキストチャット

分散環境下で行われる協調学習において、テキストチャットは発言の内容をテキストに表してから送信するため、自身の考えをまとめる

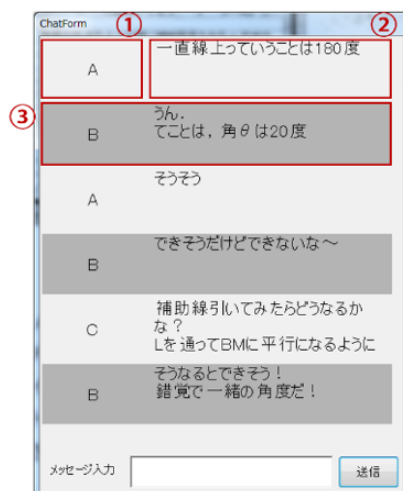


図 4.2.2 テキストチャット

ChatMessage	
msgtype	"Chat"
int	id
string	name
string	chatmsg

図 4.2.3 チャットメッセージ

作業が必要になり，対話に集中できるとして，重要な学習ツールといえる．

テキストチャット画面を図 4.2.2 に，チャットの発言内容を受信するメッセージを図 4.2.3 に示す．①発言者名，②発言内容に 1 つの AOI を定義しているため，参加者が誰のどの発言を参照しているのかわかる．チャットのメッセージは継承プロパティ `msgtype` の他，チャット固有のプロパティとして発言者の `id`，名前，メッセージ内容がある．メッセージのプロパティには発言者名と発言内容が含まれていることから，①②のように発言者名，発言内容を表示することや，発言者の `id` と自分の `id` が一致すれば③のように自分の発言の背景色を変えて表示することができる．

### 3. トーク履歴

音声認識による発言内容を残すことは，参加者の負荷なく議論を記録できる可能性がある．現在の技術では，音声技術の精度は必ずしも高くはないが，記録された内容をもとに振り返りや，議論の可視化を支援できることが考えられる．

図 4.2.4 にトーク履歴のフォーム画面を，図 4.2.5 にトーク履歴のメッセージを示す．テキストチャットと同様に発言者名を表示するラベル，発言内容が 1 つの AOI として定義されているため，ユーザが誰



図 4.2.5 トーク履歴

SpeakingMessage	
<code>msgtype</code>	"Speaking"
<code>int</code>	<code>id</code>
<code>string</code>	<code>name</code>
<code>string</code>	<code>speakmsg</code>

図 4.2.4 トークメッセージ

のどの発言を見返しているのかをシステムで認識できる。メッセージも継承プロパティ以外はテキストチャットと同様のプロパティがある。

#### 4. Web ブラウザ

Web ブラウザは協調学習において、調べ学習としての利用や、話題に対する興味内容を調べることを想定して実装した。

図 4.2.6 に Web ブラウザフォームを示す。視線検知に要する AOI を付与するために計算機にインストールされている Web ブラウザを利用せずに 1 つの学習ツールとして実装した。また、①進む・戻るボタンや、②URL の入力、③ホームボタンも実装しており、今後もブラウザとしての利便性を高めるため、タブの作成機能なども実装予定である。



図 4.2.6 Web ブラウザ

## 5. 共有ボード

共有ボードは，参加者同士で教え合いが発生する場や，図形問題などの課題設定の元，テキストで伝えられない情報を伝えるツールとして利用される．

図 4.2.7 に共有ボードの画面を，図 4.2.8 に共有ボードに誰かが書き込みを行っている最中に受信するメッセージを示す．メッセージのプロパティにはボードに書き込みをしている人の `id` があるため，複数のユーザが同時刻に書き込むことができる．また，このメッセージは 1 つ前のマウス座標( $b_x$ ,  $b_y$ ) と 現在のマウス座標( $c_x$ ,  $c_y$ ) をプロパティとして持つ．誰かが 1 つの連続した書き込みを行っている際，随時メッセージを受信し，この 2 点間の座標を細かな直線として描き続

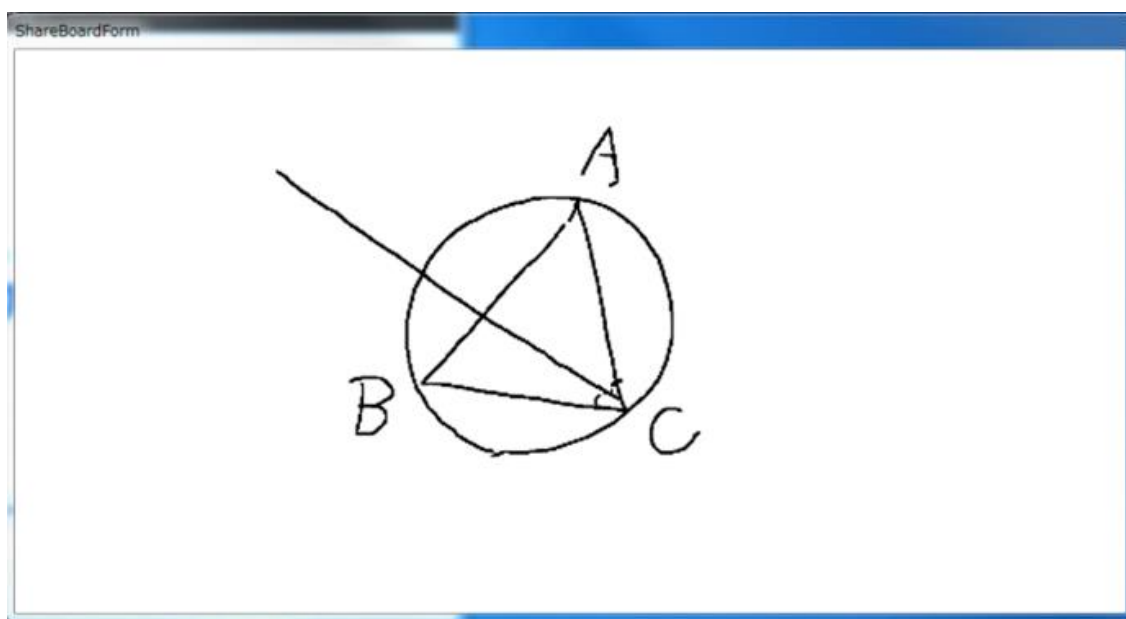


図 4.2.7 共有ボード

BoardWritingMessage	
<i>msgtype</i>	"BoardWriting"
int	id
string	name
double	cx
double	cy
double	bx
double	by

図 4.2.8 共有ボードのメッセージ

---

けることで、リアルタイムで何を描いているのか共有できる仕組みになっている。今後は誰が書いているのかの表示機能や、消しゴムなどのツールを増やす予定である。また、このボードで書き込んだテキスト内容や形を認識できる可能性もあり、今後システムが協調学習の状況を認識する際に利用できると考えている。

#### 4.3 学習ツールを活用した協調学習支援

プラットフォームに基づき言語・非言語情報を扱える学習ツールを実装した。本研究では単体の言語・非言語情報をデータレベルで通信し、実装した学習ツールに反映している。

今後、これまで協調学習支援を目的として行われてきた研究の知見を導入できると考えられる。例えば、本研究で実装した時系列順にメッセージを表示するテキストチャットに対して、スレッド構造のチャットシステム<sup>[30]</sup>を実装することで、議論でどのような話題が挙がり、各話題がどう進展したのかよりわかりやすく表示することができる。また、林ら<sup>[25]</sup>が開発した円卓場インタフェースをビデオチャットツールに導入することで、一体感が生まれやすくなり協調学習を支援することや、視線、発話区間、筆記動作を用いて学習状況を可視化するシステムの知見<sup>[15]</sup>を導入することで、学習終了後に学習状況を可視化していたが、リアルタイムに学習状況を可視化できると考えられる。

---

## 5 結論

### 5.1 本研究で得られた成果

本研究では協調学習支援システムを開発するためのプラットフォームを構築した。これまで協調学習支援システムは独自の状況に応じて開発されてきたため、学習状況に応じてカスタマイズできず、実用化できるものは少なかった。本プラットフォームの構築によって、システム開発者は学習状況に応じた学習ツールを組み入れることが可能となり、あらゆる学習状況に応じたシステムを開発できる。また、これまで言語情報を用いて協調学習を支援する研究が主流であり、非言語情報は重要な要素であるにもかかわらず、あまり研究がされてこなかった。その理由として非言語情報を分析するのに時間とコストがかかることが挙げられる。本プラットフォームではデバイスをより簡単に設置し、言語・非言語情報をリアルタイムに取得できることで、今後の言語・非言語データを用いたインタラクションの分析が発展すること、更に協調学習を支援するためのフィードバックを与えることが期待される。

### 5.2 今後の課題

今後の課題として、まずプラットフォームの利便性向上が挙げられる。システム開発者が学習ツールを組み入れる際、プラットフォーム内のプログラムを修正し、追加する学習ツール、及びそれに伴うメッセージを定義しなければならないのが現状である。今後は、新しい学習ツールやメッセージフォーマットを定義するファイルをプラットフォームに独立した形で用意する。また、プラットフォームに実装された学習ツールと言語・非言語情報のやり取りの簡便化に向けて、学習ツール単位で取得対象の情報をイベント駆動として定義・取得できる仕組みを取り入れたいと考えている。

プラットフォームに基づく具体的な協調学習支援システムを開発していくことも課題である。言語情報に加えて、非言語情報もリアルタイムに取

---

得可能な本プラットフォームに基づき、協調学習の終了後だけでなく、協調学習中にも参加者にフィードバックを与えることができる協調学習支援システムを開発したいと考えている。

---

## 謝辞

本研究は、大阪府立大学 現代システム科学域 知識情報システム学類の瀬田和久教授、林佑樹助教の丁寧かつ熱心な御指導のもとで行われました。心より感謝申し上げます。

また、日常の議論を通して多くの知識や示唆を頂いた瀬田研究室の諸氏、そして、さまざまな面でお世話になった緒方真紀専攻秘書に感謝致します。

最後に、終始暖かく見守り支えてくれた家族と友人に感謝申し上げます。

平成28年3月4日



---

## 参考文献

- [1] 文部科学省: “教育振興基本計画”, 2013.
- [2] 岡本敏雄: “協調学習環境 (CSCL): 構築と導入”, 電子情報通信学会総合大会講演論文集 1996 年.情報・システム(1), pp.389-390, 1996.
- [3] 稲葉晶子, 豊田順一: “CSCL の背景と研究動向”, 教育システム情報学会誌, Vol.16(3), pp.111-120, 1999.
- [4] 石井裕: “CSCW とグループウェアー協創メディアとしてのコンピューター”, オーム社, 1994.
- [5] P. Eisert: “Immersive 3-D Video Conferencing: Challenges, Concepts, and Implementations ”, *Proc. SPIE Visual Communications and Image Processing (VCIP)*, 2003.
- [6] R. Vertegaal: “The GAZE Groupware System: Mediating Joint Attention in Multiparty Communication and Collaboration”, *Proc. of ACM CHI 1999*, pp.294-301, 1999.
- [7] 岡田謙一, 松下温: “臨場感のある多地点テレビ会議システム : MAJIC”, 情報処理学会論文誌, Vol.36, No.3, pp.775-783, 1995.
- [8] 今井健二, 高橋由紀子, 緒方広明, 矢野米雄: “Sharlok II : WWW を利用した開放型グループ学習支援システム”, 電子情報通信学会技術研究報告, Vol.98(643), pp.111-118, 1999.
- [9] E. de Bono: “*Six Thinking Hats*”, Back Bay Books, 1999.
- [10] C. K. Looi: “Exploring the Affordances of Online Chat for Learning”, *International Journal of Learning Technology*, Vol.1, No.3, pp.322-338, 2005.
- [11] Y. Tamura and S. Furukawa: “CSCL Environment for “Six Thinking Hats” Discussion”, *Proc. of KES 2007/WIRN 2007*, pp.583-589, 2007.
- [12] 中村学, 竹内章, 大槻説平: “グループ学習支援システムにおける知的エージェントに関する研究”, 電子情報通信学会技術研究報告, Vol.ET95(14), pp.79-86, 1995.
- [13] 稲葉晶子, 大久保亮二, 池田満, 溝口理一郎: “協調学習におけるインタラクション分析支援システム”, 情報処理学会論文誌, Vol.44, No.11, pp.2617-2627, 2003.
- [14] M. J. Baker and K. Lund: “Flexibly structuring the interaction in a CSCL environment”, *Proc. of European Conference on Artificial Intelligence in Education*, pp.401-407, 1996.
- [15] 林佑樹, 小川裕史, 中野有紀子: “協調学習における非言語情報に基づく学習態度の可視化”, 情報処理学会論文誌, Vol.55, No.1, pp.189-198, 2014.
- [16] S. Goodlad and B. Hirst: “Peer Tutoring: A Guide to Learning by Teaching”, Kogan Page: London, 1989.
- [17] A. Bandura: “Social Learning Theory”, New York; General Learning Press, 1971.
- [18] J. H. Flavell: “Metacognitive aspects of problem-solving. In L. B. Resnick (Ed.)”, *The nature of intelligence*. Hillsdale, pp.231-235, 1976.
- [19] 大山牧子, 田口真奈: “大学におけるグループ学習の類型化ーアクティブ・ラーニング型授業のコースデザインへの示唆ー”, 日本教育工学会論

- 
- 文誌 37(2), pp.129-143, 2013.
- [20] 角康之, 矢野正治, 西田豊明: “マルチモーダルデータに基づいた多人数会話の構造理解”, 社会言語科学, Vol.14, No.1, pp.82-96, 2011.
- [21] Red5 Media Server: <http://www.red5.org/>.
- [22] Tobii Developer zone: <http://developer.tobii.com/downloads/>.
- [23] Adobe Flash Player: <https://get.adobe.com/jp/flashplayer/>.
- [24] 堀公俊, 加藤彰: “ファシリテーション・グラフィック 議論を「見える化」する技法”, 日本経済新聞出版社, 2015.
- [25] 林佑樹, 小尻智子, 渡邊豊英: “貢献への気づきを反映した議論支援インタフェース”, 情報処理学会論文誌, Vol.53, No.4, pp.1461-1471, 2012.
- [26] 高梨克也, 加納圭, 水町衣里, 元木環, “双方向コミュニケーションでは誰が誰に話すのか?:サイエンスカフェにおける科学者のコミュニケーションスキルのビデオ分析”, 科学技術コミュニケーション, 第 11 号, pp.3-17, 2012.
- [27] 河原達也, 李晃伸, “連続音声認識ソフトウェア Julius”, 人工知能学会誌, Vol.20, No.1, pp.41-49, 2005.
- [28] Google Speech API: <https://console.developers.google.com/project>.
- [29] airpen: <http://www.airpen.jp/mechanics/index.html>.
- [30] M. Smith, J. Cadiz and B. Burkhalter: “Conversation Trees and Threaded Chats”, *Proc. of ACM CSCW 2000*, pp.97-105, 2000.