

3D オブジェクトのためのアノテーション共有システム

今江 健悟^{1,a)} 林原 尚浩^{1,b)}

概要: 画像や動画のようなコンテンツは Web 上で無数に共有され, 検索エンジンなどを用いて必要とするものを容易に探し出すことができる. 3D プリンタの普及により, 3D オブジェクトも同様に共有される事が予想される. しかし, 画像や動画と違い, 3D オブジェクトは見ている視点位置により, 見え方が大きく変わる問題がある. そのため, 見ている視点を考慮した情報の付加が 3D オブジェクトの共有には必須である. 本研究では, 3D オブジェクトを共有することで, 今までの共有空間にはなかったカメラの視点というものに着目し, カメラの視点と他の情報を紐付ける 3D オブジェクトのアノテーション共有システムの提案とプロトタイプの実装を行う. また, Pub/Sub モデルを用いたカメラの視点を同期する機構の実装も行なっている.

キーワード: 3D オブジェクト, アノテーション, 共有空間, カメラ視点, Pub/Sub

Annotation sharing system for 3D objects

KENGO IMAE^{1,a)} NAOHIRO HAYASHIBARA^{1,b)}

Abstract: Many pictures and videos have been shared on the Web. Those are possible to easily find what you need by using a search engine. The spread of 3D printer, 3D objects are also expected to be shared in the same manner. However, unlike pictures and videos, 3D objects have a problem that the appearance changes by rotation and zoom operations. So that, the addition of annotations considering the viewpoint position is required for shared 3D objects. We focus on those of 3D objects with camera viewpoints in this work and propose a prototype implementation of the annotation sharing system of 3D objects. We also implement the synchronization mechanism for 3D objects based on Publish/Subscribe model.

Keywords: 3D-Objects, Annotation, Share Space, Camera Position, Pub/Sub

1. はじめに

1.1 背景と問題点

Web 上には無数の画像や動画といったコンテンツが共有されている. それらのコンテンツは Web 上で不特定多数の人々に共有され, 検索エンジンなどで必要とするものを容易に探し出すことができる. また, これらのコンテンツにコメントや注釈 (アノテーション) を付加して共有するという方法で, 他の閲覧者の趣向や感想などを伝えるというコミュニケーションスタイルが確立している.

その例として, ニコニコ動画 [7] などが上げられる. ニコニコ動画にはコメントを投票し, 動画の再生中に流す事により, アノテーションを付加している. これは動画を見た人がその他の人々に対して, その動画に対してのアノテーションを付加していると言える. ここで言うアノテーションとは, コメントや注釈などのテキストのことである.

3D プリンタの普及により, 2D コンテンツと同様に 3D オブジェクトも共有される事が予想される. 画像や動画というコンテンツは 2 次元の情報であるが, 3D オブジェクトは 3 次元の情報であり, 新たに共有出来る情報が増えるため, それに対応したアノテーションの表示が要求される.

¹ 京都産業大学
KSU, Kamigamomotoyama, Kita-ku, Kyoto 603-8555, Japan

^{a)} g1144155@cse.kyoto-su.ac.jp

^{b)} naohaya@cse.kyoto-su.ac.jp

1.2 研究の目的

画像や動画と違い、3D オブジェクトは見ている視点により、見え方が大きく変わるという問題がある。

例えば、正面と後ろで見え方が異なる 3D オブジェクトに対して、従来の方式で、3D オブジェクトに付けられたアノテーションを全て表示してしまうと、正面からのアノテーションと後ろからのアノテーションがまとめて表示されてしまう。これは見えているものが異なっているにもかかわらずそれらに対するアノテーションを同一に扱っていることと同義である。従って、従来の方式では、3D オブジェクトに対して、適切にアノテーションが行えているとは言えないためこれらのアノテーションは現在の視点位置に合わせて表示されるべきである。そのため、見ている視点位置を考慮した情報の付加が 3D オブジェクトの共有には必須である。

3次元の共有空間で増えた新たな情報であるカメラ視点という情報を使い、3D オブジェクトにアノテーションを付加する方法を提案する。また、Publish/Subscribe(Pub/Sub)モデルの枠組みを使い、カメラ視点を共有することで 3D オブジェクトの同期機構を実現する方式も提案する。

本研究では、上記の特徴を備えた 3D オブジェクトのアノテーション共有システムの提案とプロトタイプの実装を行う。

2. コンテンツと共有

本節では、従来の画像や動画の共有と本研究で対象としている 3D オブジェクトの共有の違いについて論じる。

2.1 2次元のコンテンツの共有

2次元のコンテンツの共有システムとは、画像や動画などの2次元の情報を共有するシステムのことである。現在最も主流な共有システムとしては、pixiv[5]、Youtube[6] やニコニコ動画 [7] などが挙げられる。検索エンジンなどでも画像検索や動画検索が行えるため、Web 自体も 2次元の共有空間であるといえる。

2次元の共有空間の特徴としては、共有する情報に直接マークアップを行えるという事である。つまり、アノテーションを付ける事ができるのである。例えば、静止画(画像)には、それぞれにアノテーションを付加し、コンテンツに対する感想が付加されて行く。動画の場合は、静止画とは違いリアルタイム性をもつため、時間軸を考慮したアノテーションが求められる。同じ 2次元のコンテンツの共有でも動画は静止画と違い、時間軸を考慮したアノテーションが求められる。静止画ではコンテンツ自身にアノテーションを紐付けるだけであったが、動画は時間軸にアノテーションを紐付けているのである。

画像や動画にはそれぞれに対応する形でアノテーションを付加することで情報の紐付けが出来ている。

2.2 3次元のコンテンツの共有

3次元のコンテンツの共有にも、2次元のコンテンツと同じくそれに対応する形でアノテーションを付加する必要がある。3D オブジェクトの場合は、アノテーションを付加する時にカメラの視点を考慮して行わなければいけない。

例えば、3D オブジェクトというのは見る方向によって画面に映っている見え方が人によって違うという状況になる。これは画像や動画などではありえなかったことである。同じタイミングで画像や動画を複数人で見てもそれぞれの目に映る見え方は同じである。各人によって見え方が違うということは起こりえない。このことに着目し、3次元の共有空間は今までになかった新たな情報である視点という情報を増えたことがわかる。

3. 関連研究

本研究では、アノテーションというものを扱っているが、アノテーションの重要性については竹原ら [1] の研究により明らかにされている。竹原らは、アノテーションと Web ページの関係についても、深く考察し、Web ページにおけるアノテーションの必要性を示した。これらの研究からも Web ページで情報を共有する場合は、アノテーションの付加を行なう事で、効果的に情報の伝達が行なえることがわかった。

また、映像に適切なアノテーションを付加することを目指した Synvie[2] がある。Synvie では、映像自体にアノテーションを付加するのではなく、シーンに対して、アノテーションを付加する事で、映像に対する効果的なアノテーションの付加を実現している。他に映像コンテンツに関連したブログコミュニティからアノテーションを獲得し、質の高いアノテーションを実現している。

見上らによるスケーラブルなストリーム Pub/Sub プラットフォームの提案と評価 [3] では、携帯端末やセンサーなどから取得できる大量の実世界情報を用いて、状況に適したサービスを提供するコンテキストウェアサービスを補助するプラットフォームを Pub/Sub モデルを用いて実現しようとしている。この論文では広告配信サービスなどで、ユーザーが店舗の半径 1km にいる時に配信するというものを、配信される広告の情報を実世界情報を用いてフィルタリングし、実現するためのルールの登録や、情報の配信を Pub/Sub モデルを用いて実現しようとしている。大量の実世界情報と広告情報の紐付けを行い、ユーザーの状況に合わせた広告の配信を実現しようとしている。

同じく、実世界情報を紐づけることで情報を有効に利用したものとして北田らの無線アドホックネットワーク上のクチコミ情報流通サービスに関する研究 [4] が挙げられる。この研究では、インターネット上に公開された地域向け情報の不均衡性を解消した情報流通サービスを提案している。また、投票されたクチコミ情報に位置情報とカテゴリ情報

を付加している．これにより，位置情報をもとに地域のグループを構成する．カテゴリ表示条件が和食だとすると，位置情報を付加していなければ，北海道の和食店と九州の和食店が表示されてしまうため，これらを防ぐために，地域の情報というのを位置情報を元にフィルタリングする．

4. 3D オブジェクトのためのアノテーション共有システム

本研究では，3D オブジェクトのためのアノテーション共有システムを提案する．このシステムは，2つの特徴を持っている．

1つ目は，現在のカメラ位置に合わせたアノテーションを表示する仕組みである．表示すべきアノテーションに位置情報を付加し，カメラの視点位置を元にアノテーションをフィルタリングする．これによってカメラの視点位置に合わせた適切なアノテーションを表示することができる．

2つ目は Pub/Sub モデルを用いた，視点情報の同期機構の提案である．他人の見ている視点をリアルタイムに共有したい場合にこの手法を用いることで，ある人の視点情報を元にしたコミュニケーションが行える．同期機構は基本的には重い処理であるが，この手法は Pub/Sub モデルを用いて，スケーラビリティを損なう事なく，視点の共有を行うことができる．

4.1 操作について

本システムでは，回転とズームによって 3D モデルの操作を行う．回転とは，原点からの距離を変えずに，原点からの角度を変化させる操作のことである (図 1)．

回転は $rotate(\theta, d)$ で表され， θ は現在の原点からの角度， d は変化させる角度の大きさである．回転は主に，上下左右の移動の時に用いる操作である．オブジェクトの見る位置を変える時は回転の操作を行うものとする．原点からの距離 l を変化させずに，原点からの角度 θ だけを変更する操作を回転と定義する．

ズームとは，回転とは逆に原点からの距離 l を変化させる操作のことである (図 2)．

ズームは $zoom(l, d)$ で表され， l は現在の原点からの距離， d は原点からの距離の変化量である．ズームは，見ている視点の奥行きを変化させる場合に用いる操作である．

3D オブジェクトの表示は，上記の操作を用いて行われるものとする．

4.2 位置情報付きアノテーション

前述の通り，3D オブジェクトは見ている位置により，3D オブジェクトの見え方が異なるため，従来のアノテーション方式をそのまま適応できないという問題がある．この問題を位置情報付きアノテーションを導入する事で解決する．

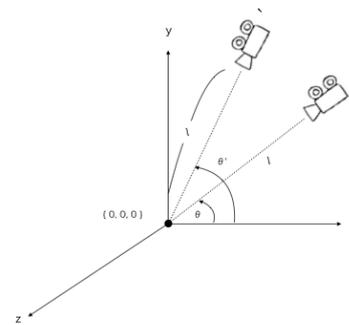


図 1 回転の定義

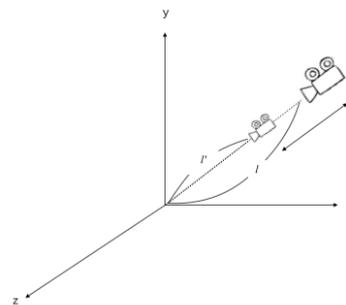


図 2 ズームの定義

4.2.1 利点

アノテーションに，それを付加した位置情報を属性として持たせる事で，アノテーションの表示と回転やズームといった操作と関連づける．

例えば，左右の見え方がまったく違う見え方になる 3D オブジェクトの場合は，左用のアノテーションと右用のアノテーションを分ける事で適切にアノテーションを付ける事が出来る．更に，3D オブジェクトのどこに対してつけたアノテーションなのかわかるようになるため，3D オブジェクトのある部分に対してアノテーションをつけるといったような使い方もできるようになる．これは，自分の視点を間接的に閲覧者と共有させているとも考える事が出来る．

4.2.2 アノテーションの定義

位置情報付きアノテーションを実装する場合には，アノテーションの集合を定義する必要がある．アノテーションの集合の定義を以下に述べていく．

$$X = (C_1, C_2, C_3, C_4, \dots, C_n) \quad (1)$$

アノテーションの集合は X で表され， C_n という要素によって表される． C_n は以下の要素で構成される集合である (文字列: s ，位置情報: (x, y, z))

$$C_n = (s, (x, y, z)) \quad (2)$$

アノテーションの集合は 3次元空間上の集合であると考えられる．上記の定義を図に表したものが図 3 である．

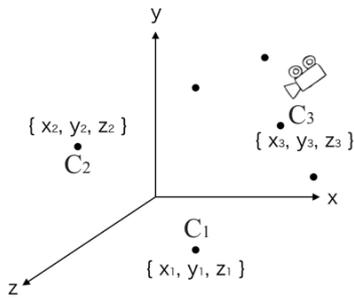


図 3 アノテーションの集合

アノテーションの集合は図 3 から分かるように 3 次元空間上に配置される。また、新たなアノテーションを追加する場合は以下の定義を満たすように行う。

$$X' = X \cup C_i \quad (3)$$

X' は現在のアノテーション集合 X と新たに追加されるアノテーション C_i との和集合である。この定義が満たされるようにアノテーションを追加しなければならない。

4.2.3 アノテーションの選別

アノテーションの定義については既に述べたため、これから表示すべきアノテーションを選別する方法について述べていく。3 次元空間上のカメラ視点の位置を元に現在表示すべきアノテーション集合のフィルタリングを行う。 x, y, z 座標軸上のあるアノテーション c のカメラ位置からの距離を返す関数をそれぞれ $f_x(c), f_y(c), f_z(c)$ とする。

上記で定義した関数を用いて、表示すべきアノテーションの集合を定義する。表示すべきアノテーションの集合 D というのは以下の式を満たすものである。

$$D = \{c \in X \mid f_x(c) \leq r, f_y(c) \leq r, f_z(c) \leq r\} \quad (4)$$

上記の式は、表示するアノテーションはカメラ視点の位置を中心とした半径 r の円に含まれるアノテーションの集合 (図 4) を意味している。

図 4 のような選別方法を行っている理由は、ある位置から見た時の視点位置が半径 r 以内ならば、似た見え方になることが予想されるからである。 r が適切な値になっている場合、その位置から見た見え方に対する適切なアノテーションが表示されていると言える。選別領域というのは、アノテーションの表示されるべき誤差範囲であるともいえる。アノテーションが投稿された位置と同じ位置に来ない限り表示されないとすると、アノテーションが表示される可能性が極めて低くなる。表示誤差を設ける事で、従来方式と同様の弾幕を貼るようなアノテーションの表示が可能である。3 次元空間上をカメラが移動する事を考えると、ある程度の誤差を許す必要があるのである。

4.2.4 アノテーションの表示

アノテーションの選別は前述の方法を行うことでうまくいくが、アノテーションの持っている位置情報は 3 次元座

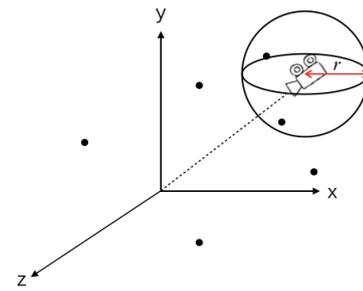


図 4 アノテーションの選別

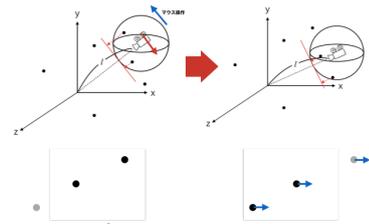


図 7 アノテーションの表示の変化

標なので、そのままでは画面に表示できない問題がある。つまり、3 次元の位置情報を 2 次元の位置情報に変換する必要がある。3 次元の位置情報を 2 次元の位置情報に変換する関数を $M(x, y, z)$ とする

それぞれのアノテーションの位置情報 (x_i, y_i, z_i) を現在のカメラ位置 (x, y, z) を中心として平行移動し、 $M(x, y, z)$ を用いて 2 次元座標に変換します (図 5)。この方法の利点としては、現在のカメラ位置からの位置関係を保持したまま変換することができる。例えば、カメラ位置から右にあるアノテーションは画面に表示した時に、画面の右側に表示されます。

更にこの変換のもう 1 つの利点として、3D オブジェクトの見ていた視点を変更した場合にアノテーションの位置関係がうまく更新される特徴がある。例えば、先ほどの図 5 から図 6 の状態に移動した場合の変化を図 7 に示している。図 7 は現在の位置から右方向に回転した場合の例である。右方向に回転したように見せる場合はカメラ位置は反対の左方向に回転することとしている。この図 7 からわかるように、画面に表示されているアノテーションは回転した方向に移動している。このようにアノテーションも回転に合わせて位置が変わるという表現を今回の提案手法を採用する事で、同時に実現している。

しかし、そのままでは画面に図 7 のように表示することができない (図 8)。3 次元空間上のスケールと画面サイズのスケールに大きな違いがある場合がある。この場合、3 次元空間上の大きさは数値的には小さな値となることが多いが、画面のサイズは逆に大きな値になりがちである。この問題を考えずに、実際に表示してみると図 8 のようになってしまう。この問題の解決策として座標の補正を行な

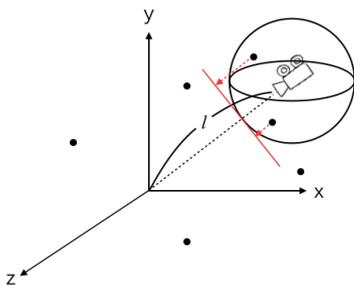


図 5 回転によるカメラ位置の変更とアノテーションの表示 1

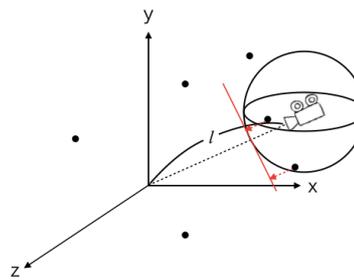


図 6 回転によるカメラ位置の変更とアノテーションの表示 2

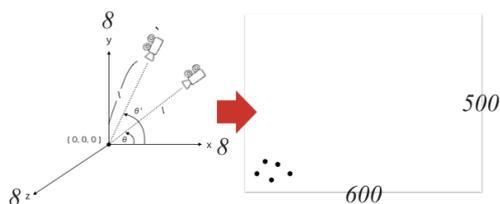


図 8 アノテーションの表示時のスケールの問題

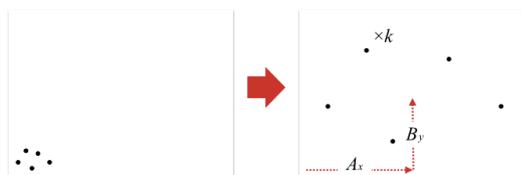


図 9 スケールの問題の解決策

う関数 $Comp$ を導入する .

$$Comp(x, y) = (A_x + kx_i, B_y + ky_i) \quad (5)$$

2次元座標に変換した位置情報 (x_i, y_i) を補正する事で、図 9 のように表示する事が出来る . A_x は横幅の最大値の半分、 B_y は縦幅の最大値の半分、 k は補正定数である . 図 9 のように補正する事で、それぞれの空間のスケールの違いを解決する .

5. Pub/Sub モデルによる同期機構

2つ目の提案手法は、Pub/Sub モデルの枠組みを利用する事で、3D オブジェクトの同期機構を実現できるというものである . Pub/Sub モデルを用いて、視点情報自体をそのまま Publish するというを行う . これは、ある閲覧者が前述の操作 (図 1, 図 2) を行った場合に現在の視点情報、つまりカメラ位置を Publish するという方法になる . 視点情報自体を共有する事で同期機構を実現できる . 他人のしている視点をリアルタイムに共有したい場合にこの手法を用いることで、ある人の視点情報を元にしたコミュニケーションが行える .

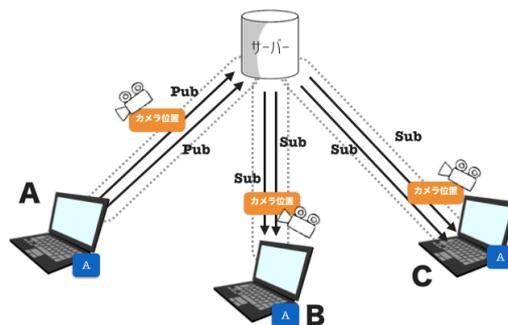


図 10 Pub/Sub モデルによる視点の共有

上記のデータのやり取りを図にしたものが図 10 である . 図 10 を見てもらえるとわかると思うが Pub/Sub モデルの形をそのまま当てはめて実現している . A の視点が欲しい場合は、A のトピックを登録する . すると A が視点を Publish するとそれを Subscribe しているユーザに視点情報が送られ同期処理が行われる .

配信側が視点情報を見てもらいたい人を意識せずに行う画面同期といえる . これは、Pub/Sub の枠組みを適用するだけで実現出来ることが優れている . 共有したい人はその人の視点情報を購読すればよいだけとなりお互いに疎通確認などを行う必要がないため、スケーラブルなシステムを構築出来るという特徴がある .

5.1 Pub/Sub モデルについて

Pub/Sub モデルとは、出版-購読型モデルと言い、メッセージの出版者 (Publisher) が特定の購読者 (Subscriber) を想定せずにメッセージを送る非同期メッセージングモデルである . Publisher はトピックと呼ばれる属性情報をメッセージに付加し、サーバに送信する . この操作は Publish と呼ばれる . Subscriber は自分が興味のあるトピックを登録することができる . この操作を Subscribe という . Subscriber はメッセージが Publish されて来た時に、登録されているトピックであれば取得する . Publisher と Subscriber の結合度が低い為、スケーラビリティが高いシステムを構築出来る . 本研究で上記の特性を考え、視点共有に Pub/Sub モデルを採用出来ると考えた .

表 1 開発環境

種類	バージョン
Node.js	v0.11.10
Redis	v2.6.14
Express	v3.4.8
Socket.io	v0.9.16

5.2 利点

同期処理を実装する場合は、一般的にコストの高い処理になりがちであったが、Pub/Sub モデルを適用する事でスケラブルな同期処理を実現する事ができる。

このシステムの応用としては、例えば医療分野の手術を行う前の遠隔ミーティングなどが考えられる。説明を行う人が操作を行い、手術対象（臓器など）の3Dオブジェクトを動かしながら手術の予定を伝えるというように、どこを見れば良いのかを視点情報の共有によって実現できる。これは聞き手が話し手の視点情報を購読するモデルの一例である。

この例では、従来の同期システムと違い、配信側は視点情報を Publish する。閲覧者側は登録 (Subscribe) しているユーザの視点情報を受信するだけで実現できるため、非常に実装のコストも少ないと言える。

6. 実装の詳細

本研究で実装した、Pub/Sub モデルに基づく3Dオブジェクトのアノテーション共有システムについて、詳しい実装内容について紹介する。開発環境は表1である。

また、本システムの構成図を図11に記す。本システムはNode.jsを基盤にWebブラウザ上で動くシステムとして開発しました。実行環境は、MacOSXバージョン10.9.5を使用し、WebブラウザはSafariのバージョンは7.1を使用しました。本システムはNode.jsのWebアプリケーションフレームワークExpressとWebSocket関連のモジュールであるSocket.ioを用いて実装した。

視点情報をサーバ側から、自動的に受け取るため、サーバとの通信手段はWebSocketを採用している。WebSocketは従来までのHTTP通信とは違い、サーバからのPush通信を実現している。サーバに視点情報を送信するとその視点に合わせた情報をWebブラウザの更新処理を必要とせずにWebブラウザ上に反映することが出来る。WebSocketにより双方向の通信を実現することによりアノテーションの表示位置の更新やPub/Subによる視点同期の処理を実現している。

3Dオブジェクトの表示に関しては、WebGLを用いた。WebGLはWebブラウザで3DCGを表示させるための規格である。本研究ではWebGL自体を使用するというよりWebGLのラッパーライブラリのThree.jsをメインに使用した。

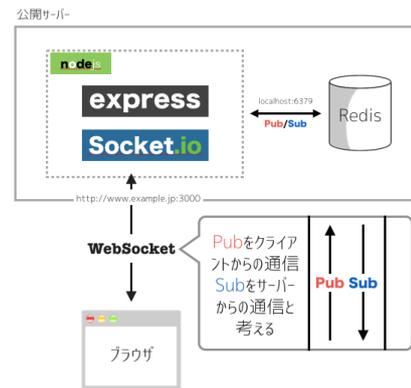


図 11 本システムの構成図

そして、データベースはRedisというKVSを使用している。Pub/Sub通信をサポートしているKVSであるためデータベースはRedisを採用した。図11からわかるように、Redisとの通信はPub/Sub通信を使い接続しているため、クライアント側からの通信をPub、サーバ側からの通信をSubと考え通信することで、クライアントがイベントを起こした場合に即座にデータベースに反映され、サーバからのPush通信のタイミングをSubで実現している。上記のようなWebSocketとPub/Subモデルを用いた方法で現在の視点情報を用いたリアルタイムな処理を実現した。

7. まとめと考察

本論文では、3Dオブジェクトのための位置情報付きアノテーション共有とPub/Subモデルを用いた視点情報の同期機構の提案、およびそれらのプロトタイプの実装について述べた。

3Dオブジェクトはそれ自体を回転、ズームなどの操作によって動かす事により、見え方が変わってしまうという問題点があった。この問題をアノテーションに視点情報を紐づけることにより、3Dオブジェクトが持つ見る方向によって画面に映っている見え方が人によって違うという特性を生かしたままアノテーションを付加する方法を提案した。これにより、3Dオブジェクトに関連付けられたアノテーションを付加し、それらを共有するユーザに適切に表示するシステムを構築する事ができた。

Pub/Subモデルを用いた視点情報の同期機構は、3次元の共有空間から新たに増えた視点という情報をPub/Subモデルの枠組みに当てはめる事で同期処理が実装できることが特徴である。具体的には、視点位置を出版し、視点共有する側のユーザはその位置情報を購読することで3Dオブジェクトの同期を行なう。このPub/Subモデルを用いた3Dオブジェクトの同期は従来手法と比べて、実装の容易さとスケラビリティに優れている。

本研究で提案したシステムは、3Dオブジェクトに基づいたコミュニケーションなどを行なう際に有用である。例

えば、医療分野において手術を行う前に対象となる臓器の3Dオブジェクトを用いた遠隔ミーティングなどが考えられる。

今後は、3Dオブジェクトのアノテーション共有、視点共有に加えて、複数のユーザ間で協調編集を行なう機能を実現し、3Dオブジェクトのための統合プラットフォームをめざす。特に、Pub/Subモデルに基づいた協調編集には、通常排他制御などのコストのかかる処理が必要であるが、CRDT(Commutative Replicated Data Type)[8]に基づいたロックフリーな実装を行なう事により、この問題の解決を試みる。

参考文献

- [1] 竹原幹人, 田中浩也, 角谷和俊, 田中克己, Webコンテンツとアノテーションコンテンツ間の動的関連付け及びその相互作用, 情報処理学会研究報告, 17-23, 2003.
- [2] 山本大介, 増田智樹, 大平茂輝, 長尾確, 映像を話題としたコミュニティ活動支援に基づくアノテーションシステム, 情報処理学会論文誌 48(12), 3624-3636, 2007.
- [3] 見上紗和子, 佐藤正, 落合敏功, 喜田 弘司, スケーラブルなストリーム Pub/Sub プラットフォームの提案と評価, 電子情報通信学会技術研究報告, 2011.
- [4] 北田高之, 矢倉健一郎, 加藤祐介, 高木由美, 太田 能, 西本大介, 松井亮太, 山口和泰, 桑野満博, 無線アドホックネットワーク基盤における Pub/Sub 手法を用いたクチコミ情報流通サービスの構築, 電子情報通信学会総合大会講演論文集, 2008.
- [5] ビクシブ株式会社, イラストコミュニケーションサービス [pixiv(ビクシブ)], 入手先 (<http://www.pixiv.net>), (2014/11/14 アクセス).
- [6] チャド・ハーリー, ステイブ・チェン, ジョード・カリム, YouTube, 入手先 (<https://www.youtube.com>), (2014/11/14 アクセス).
- [7] niwango.inc, ニコニコ動画, 入手先 (<http://www.nicovideo.jp>), (2014/11/14 アクセス).
- [8] M. Letia, N. Preguiça, M. Shapiro, "CRDTs: Consistency without Concurrent Control", Research Report, INRIA/RR-6956-FR+ENG.