

空間情報処理のオフローディングによる リアルタイムAR積雪可視化システムの検討

小林 靖明¹ 川上 朋也¹ 松本 哲² 義久 智樹² 寺西 裕一^{3,2}

概要：

降雪が多い地域における住民や訪問者の行動支援を目指し、現在地の将来の積雪状況を視覚的に把握可能とするリアルタイムAR積雪可視化システムを提案する。提案システムは、3D LiDARを用いてセンシングした現実世界の空間情報に積雪状況を反映した積雪空間情報を生成し、カメラで撮影された映像に重畳表示する。本稿では、処理性能が小さいモバイルデバイスに対応可能とするため、処理量が大きい積雪空間情報の処理をモバイルデバイスからエッジサーバへオフローディングするシステム構成による提案システムの実現法について述べる。

1. はじめに

降雪が多い地域において、円滑かつ安全に生活を送る上では、予想される積雪量を常に把握しておくことが重要である。降雪が多い地域では、雪による渋滞などの対策で多くの公道に消雪装置が設置されるが、狭い路地、住宅前の通路、屋外駐車場等に積もった雪は、住民が自ら雪かきを行い、除去する必要がある。また、雪かきが行われていない場所を移動する際には、積雪量に応じて撥水性のあるブーツや衣服を着用するなどの対策が必要となる。気象庁は、当日6時から18時までの12時間降雪量（平均値、最大値）の予測を1日2回発表している。この情報を参照すれば、経験の長い地元住民は積雪量をある程度予測することができる。しかし、観光客や訪問者は降雪によってどの程度の積雪が起きるか、また、街がどのような状態となるかをイメージすることは難しい。さらに、地元住民であっても、過去に経験したことのない降雪量が予測された場合や、自宅以外の場所に居る場合、積雪量をイメージすることは難しくなる。

本研究では、Augmented Reality (AR) 技術を活用し、モバイルデバイス上でカメラ映像に積雪が起こった状態を重畳表示させ、積雪状態を視覚的に表現するリアルタイムAR積雪可視化システムを提案する。提案システムは、現

在カメラに写っている景色に、予測される降雪によって、以後、どのように雪が積もるかをAR表示することで、積雪の量を感覚的に把握可能とする。

上記システムの実現には、予想される積雪状況を反映した積雪空間情報を生成し、モバイルデバイス上で重畳表示する必要がある。これまでに、センサー情報や気象現象をAR表示するシステムの研究開発が進められてきた [1][2] が、現実世界に重畳して積雪空間を視覚的に表現可能なARシステムは見当たらない。積雪空間の生成には、センシングされた現実世界の空間情報に積雪状態を反映させる必要がある。こうした処理は膨大量の空間データを扱わねばならず、モバイルデバイス上で全ての処理を完結することは難しい。提案システムは、この課題に対処するため、研究開発や標準化が盛んに進められているエッジコンピューティング技術 [3], [4] を適用し、積雪空間情報の生成処理をエッジサーバへオフローディングするシステム構成・設計とした。

本稿では、提案するリアルタイムAR積雪可視化システムの実現方法について述べる。以下、2章では、関連研究について述べる。3章では、提案システムの設計について述べ、4章にて、提案システムの実装および評価結果について述べる。最後に、5章にてまとめと今後の課題について述べる。

2. 関連研究

積雪に関して、多数の研究が行われている [5], [6], [7]。文献 [5] では、屋根に積もった雪を処理する際の事故の軽減を目的として、「雪おろシグナル」というシステムを開

¹ 福井大学大学院工学研究科
Graduate School of Engineering, University of Fukui

² 大阪大学サイバーメディアセンター
Cybermedia Center, Osaka University

³ 国立研究開発法人情報通信研究機構
National Institute of Information and
Communications Technology



(a) 非積雪時 (b) 積雪時

図 1 非積雪時と積雪時の状況

発している。この「雪おろシグナル」は、積雪重量の分布を地理院地図上に表示することで、屋根雪おろしの判断材料に用いることが出来る。また、文献 [6] では、Web 上でリアルタイムに公開される複数の機関の積雪観測データを収集・統合し、積雪深の空間分布とその時間変動を自動的に解析するシステム「準リアルタイム積雪分布監視システム」の開発を行っている。「準リアルタイム積雪分布監視システム」では、Web 上で提供される膨大な数の積雪観測データについて、Web スクレイピング技術を用いて収集し、積雪深および降雪量の空間分布を自動的に解析し、分布図を生成している。しかし、これらの研究は、地図上に積雪重量を表示しており、実際の積雪をイメージすることは難しい。文献 [7] では、積雪深を計算するモデルの構築・検証を行っている。このモデルは、降水量、気温などの気象データに基づいて構築されており、融雪量計算や雪の密度を考慮することで、より正確な積雪深を算出している。しかし、この研究は、数値的に積雪深を把握することは可能であるが、実際に積雪をイメージすることは難しい。

積雪状態の表示に関する研究として、積雪の形状をリアルタイムでモデリングする研究がある [8]。文献 [8] では、積雪のアルゴリズムを簡略化することで高速にモデリングする手法を提案している。上記は 3D 空間上に積雪状態の表示に資する技術であるが、本研究が目指す、現実空間に積雪状態を AR 表示するものではない。

センサー情報を AR により可視化するシステムとして uMegane [1] がある。uMegane はビジュアルマーカーを用いて現実世界の映像中のマーカー部分にセンサーから得られたセンシング情報を表示する。AR お天気シミュレーター [2] は、スマートフォン上で、河川が氾濫した場合の浸水状況をカメラでとらえた景色に重畳表示する機能を提供する。また、雪が降る様子を重畳表示する機能も持つ。上記システムは、いずれもリアルタイムの AR システムであるが、現実空間への積雪状態の重畳表示を可能とするものでない。AR お天気シミュレーターが持つ浸水状況の可視化機能は、一定の高さに水面を一律に表示する機能で

あり、現実世界の形状に応じた積雪状況の表示が必要となる積雪の可視化には利用できない。

3. 設計

本章では、我々が提案するリアルタイム AR 積雪可視化システムの設計について述べる。

3.1 AR システムとしての要件

提案リアルタイム AR 積雪可視化システムは、モバイルデバイスの画面内に、カメラで撮影した現実世界の映像に対して積雪状況を重畳させた映像を表示する。

予測される積雪の高さは、気象状況や位置に応じて変化する。住民や訪問者が視覚的に将来の積雪状況を把握可能とするには、モバイルデバイスの位置に合わせた積雪予測情報を取得し、AR 表示に反映させる必要がある。

図 1 は、実風景における非積雪時と積雪時の状況の例である。例えば、道路、階段、ベンチ、樹木等には、降雪量に応じて積雪が発生した状態となる。一方、壁面や電柱などには積雪が発生しない。また、形状や大きさによっては、物体として存在しても雪が積もらないこともある。このように、積雪表示は、映像に映った実世界にある物体の大きさ、形状、状況に合わせて行う必要がある。

AR システムとしては、モバイルデバイスのカメラ視野の変化に追従し、映像に映った現実世界に積雪状況が重畳された映像が表示される必要がある。撮影される映像の画角は、モバイルデバイスのアングルや向きに応じて変化するが、この画角に対して重畳される積雪の表示がリアルタイムに追従できることが望ましい。

3.2 積雪状況の表示

本研究では、現実世界の積雪状況を、ポリゴンメッシュの積雪オブジェクトとして表現するものとする。ポリゴンメッシュとは、オブジェクトの形状を多面体として定義したときの頂点、面の集合である。本研究では、三角形の集合によるポリゴンメッシュを用いる。空間あたりの三角形数を多くすれば、より精緻な形状の積雪オブジェクトが表現可能となるが、積雪オブジェクトのデータ量は増加する。一般に、雪は、垂直、あるいはそれに近い物体の部位や、上方に物体がある領域には積もらない。従って、積雪オブジェクトの形状は、積雪が発生する空間を判定した上で決定する。

積雪オブジェクトの高さは、降雪量に応じて増加する。降雪量は、気象庁が公開する降雪予報情報をもとに得ることができる。本研究では、モバイルデバイスから GPS (Global Positioning System) による現在位置情報に基づいて、現在地に応じた降雪予報情報を取得し、積雪オブジェクトの高さを得るものとする。

雪には粘り気があるため、降雪にともない土台となるも

のよりもやや広い位置に丸みを帯びた形状で積もる。この状態は、冠雪(かむりゆき)とも呼ばれる。降雪量によっては、この冠雪状態を考慮した積雪オブジェクトの形状決定が必要となる。

3.3 現実世界の空間情報

上記積雪オブジェクトの生成、および、積雪オブジェクトの重畳表示のためには、現実世界の空間情報を取得する必要がある。

本研究では、積雪オブジェクトの生成に必要な現実世界の空間情報を、三角形の集合によるポリゴンメッシュとして取得する。

空間情報は、 (x, y, z) で表現される3次元の空間座標を要素とし、三角形の頂点ベクタ $\{v_1, v_2, v_3\}$ の集合 T によって表現する。 T の各要素は、現実世界の特徴点に対応し、三角形の構成を示す頂点の組み合わせ $(v_1, v_2, v_3) \in T$ は、 v_1, v_2, v_3 それぞれがある座標 $(x_{v_1}, y_{v_1}, z_{v_1}), (x_{v_2}, y_{v_2}, z_{v_2}), (x_{v_3}, y_{v_3}, z_{v_3})$ と対応している。

本研究では、ポリゴンメッシュの取得方法は限定しない。例えば、Apple社のスマートフォンであるiPhone[9]は、精度の高い上記構造のポリゴンメッシュを取得することが可能な3D LiDAR (Light Detection And Ranging) センサーを搭載している。三角形の辺を共有する三角形の頂点は重複することになるため T の各要素 $t \in T$ は座標構造データへのポイントのベクタとして保持される。

3.4 積雪オブジェクトの生成処理

積雪オブジェクトを生成するアルゴリズムを Algorithm 1 に示す。

Algorithm 1 は、センサーによって得られた現実世界の空間情報 T 、および、積雪量(高さ) h を入力とし、積雪オブジェクト T' を出力する。本アルゴリズムでは、 T に含まれる三角形と、水平面とのなす角を求め、予め与えられた閾値 α ($0 < \alpha < 90$) を用い、求めた角度 θ が $90 - \alpha < \theta < 90 + \alpha$ を満たす三角形の座標と、その組み合わせのみから成る三角形と頂点座標の集合として T' を得ている。 α は積雪が起る土台とならなくなる角度の閾値である。厳密には α は雪質や現実世界にある物体の材質等によって調節が必要となる。

3~5 行目は、入力である特徴点座標と三角形の頂点から、一つの三角形の各頂点の座標をそれぞれ取得している。6,7 行目では、三角形の2辺に対応するベクトルを得ている。8 行目は、6,7 行目で計算した2つのベクトルの外積を計算することで、法線ベクトルを取得している。9 行目は、求めた法線ベクトルと、 xz 平面の法線ベクトル $(0, 0, 1)$ とのなす角度を、余弦の逆三角関数を使用して求めている。10,11 行目では、求めた角度が一定角度より傾いていないとき、その三角形を T' に追加している。この時点で得ら

Algorithm 1: 積雪オブジェクト生成処理

Input: T : 三角形の頂点座標 (v_0, v_1, v_2) の集合,
 h : 積雪量
Output: T' : 積雪オブジェクトの三角形頂点座標の集合

```

1  $T' \leftarrow \emptyset$ 
2 for  $t \in T$  do
3    $v_0 \leftarrow t[0]$            ▷ 頂点  $v_0$  の座標ベクトル
4    $v_1 \leftarrow t[1]$            ▷ 頂点  $v_1$  の座標ベクトル
5    $v_2 \leftarrow t[2]$            ▷ 頂点  $v_2$  の座標ベクトル
6    $vv_0 \leftarrow v_1 - v_0$ 
7    $vv_1 \leftarrow v_2 - v_1$ 
8    $v \leftarrow vv_0 \times vv_1$      ▷ 三角形の法線ベクトル
9    $\theta \leftarrow \frac{180.0}{\pi} \times \arccos\left(\frac{v[1]}{|v|}\right)$ 
10  if  $\neg(90 - \alpha < \theta \wedge \theta < 90 + \alpha)$  then
11     $T' \leftarrow \{T' \cup t\}$ 
12 for  $v' \in \text{coordinates}(T')$  do
13    $v'.y \leftarrow v'.y + h$      ▷ 積雪量に合わせ  $y$  座標調整
14 ( $T'$  の形状調整 (三角形の追加) )
15 return  $T'$ 

```

れた T' は、積雪が発生する土台に相当する平面に対応するポリゴンメッシュ情報である。12,13 行目は、積雪量に応じて積雪オブジェクトの y 軸方向(縦方向)の位置を調整している。 $\text{coordinates}(X)$ は、三角形の集合 X から抽出された重複のない頂点座標の集合を表す。14 行目は、積雪オブジェクトに対し、形状調整のために三角形を追加する処理である。この形状調整処理によって、 T' に対し冠雪に伴う形状の修正や、側面に相当する頂点の追加などを行う。

3.5 システム構成

上記で述べた積雪オブジェクトの生成処理は、現実世界の空間情報に基づいて行う必要がある。モバイルデバイスを用いたセンシングにより、空間情報(ポリゴンメッシュ情報)のデータ量は増加していくため、必要となるデータ領域の容量、ならびに、計算量は増大する。一般に、モバイルデバイスの性能は限定されており、空間情報に関する処理を全てモバイルデバイス上で完結させることは難しい。このため、本研究では、近年盛んに研究開発が進んでいるエッジコンピューティング環境を想定し、処理の一部をエッジサーバへオフローディングする構成によるシステム設計をおこなった。

リアルタイム AR 積雪可視化システムの処理のオフローディング方法として、モバイルデバイスで撮影された映像に合わせた積雪オブジェクト重畳後の映像をサーバ上で合成し、モバイルデバイスは合成された映像を表示するのみとする方法が考えられる。しかし、この方法では、映像をネットワークを介して送受信するオーバーヘッドが生じるため、描画速度(fps)が低下してしまう可能性がある。

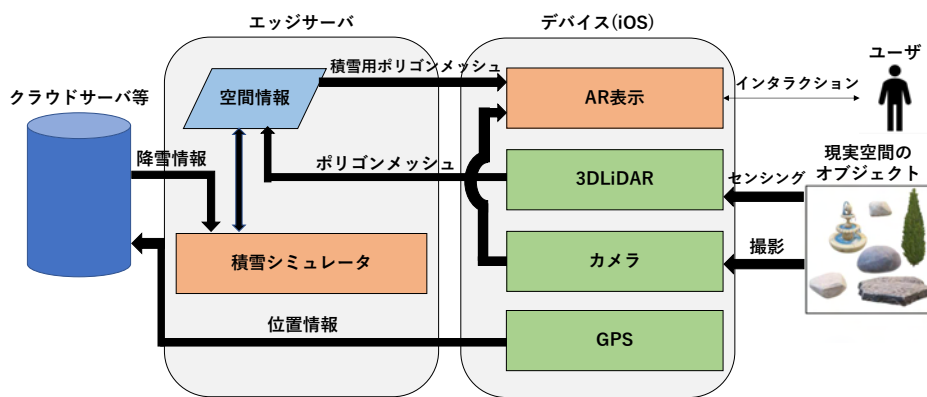


図 2 システム構成



図 3 積雪表示の例

そこで本研究では、AR 表示する映像の生成はモバイルデバイス上で行うものとし、上記で述べた積雪オブジェクトの生成処理をエッジサーバへオフローディングする構成とした。エッジサーバは、モバイルデバイスでセンシングされた実空間のポリゴンメッシュ情報を受信し、モバイルデバイスの画角内に表示が必要となる積雪オブジェクトを生成する。モバイルデバイス上では、生成された積雪オブジェクトをエッジサーバから受信し、ポリゴンメッシュのレンダリングを行った上で実空間の撮影映像に対して重畳表示する。

図 2 は、提案システムの構成を示している。本システムは、モバイルデバイスとして、LiDAR センサー、カメラ、GPS (Global Positioning System) を搭載するデバイスを想定する。Apple 社の iPhone 12 Pro, 13 Pro 等の iOS デバイスはこのシステム構成の要件を満たす。

エッジサーバは、LiDAR センサーから得られるポリゴンメッシュ情報のほか、GPS の位置情報を受信し、外部のクラウドサーバ等からモバイルデバイスの位置に対応する降雪情報を取得する。図中の「積雪シミュレータ」は冠雪情報の計算等を含む降雪オブジェクトの形状調整を、積雪シミュレーションによって行うモジュールである。本モジュールの詳細は現在検討中である。

4. 実装および評価

本章では、提案システムの実装および初期評価について述べる。

4.1 実装

モバイルデバイスとして iOS デバイスを想定し、AR アプリケーション開発を Apple Xcode 12, Swift 5 を用いて、3 章で述べたシステム設計に基づく実装を行った。ポリゴンメッシュの AR 表示には、iOS が有する ARKit[10] の機能を用いている。エッジサーバとして CPU 2.7GHz クアッ

ドコア Intel Core i7, メモリ 16GB を搭載する MacBook Pro を用い、積雪オブジェクトの生成処理は Java 11 を用いて実装した。

エッジサーバ、モバイル間は、アクセスポイントを経由せず、IEEE 802.11ac による Wi-Fi で直接接続した。また、データ交換には、HTTP を用いた。

iOS では、LiDAR センサーによる点群のセンシングは、アンカーと呼ばれる $2m \times 2m \times 2m$ の立方体ごとに行われる。アプリケーションプログラムは、アンカー毎に点群データを取得する。点群データとして得られる座標は、アプリケーション起動時のデバイスの位置を原点とし、デバイスの画面手前側を z 軸の正の方向とする右手座標系となる。エッジサーバとの間の通信は、画面上の AR 表示処理と非同期に、バックグラウンドで行われる。本実装では、アンカー毎にデータが得られる度にエッジサーバへデータを送信し、生成された積雪オブジェクトを受信して、ARKit で表示する動作とした。

iPhone 13 Pro Max 上での本システム動作時の画面キャプチャを図 3 に示す。図のように、LiDAR センサーによる点群のセンシングが進むに従い、画面上に積雪オブジェクトが AR によって重畳表示される。図 3 では、 $h = 0$ の積雪オブジェクト (表面に薄く積雪している状態) の表示を示している。

4.2 評価

提案システム実装の初期評価として、オフローディングにかかる通信オーバーヘッド (通信データ量) を計測した。

撮影映像と、オフローディングのために送信された空間情報 (ポリゴンメッシュ) のデータ量の時間経過に伴う変化の例を図 4 に示す。

図 4(a) は階段を映したものであり、時間経過とともに画角を少しずつ変化させた。図 4(c) は植木を映したものであり、映す場所はほとんど変化させていない。図 4(b), (d)

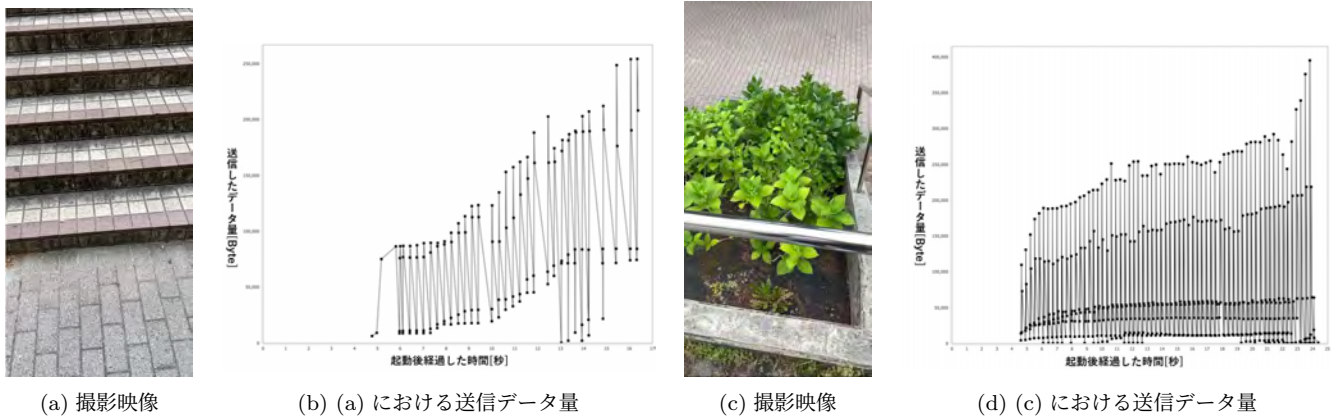


図 4 撮影映像と送信データ量

はそれぞれにおける送信データ量の時間経過に伴う変化を示している。

結果から、画角が変化しセンシングが進むに従い、送信するデータ量も増加している。これは、一つ一つのアンカー ($2m \times 2m \times 2m$ の立方体内) の映る箇所が増えることで、データ量が増加していると考えられる。また、撮影された範囲が広がることで、それまで映っていなかった $2m \times 2m \times 2m$ の立方体の範囲が映り、新たに別の空間情報データが送信されている。

エッジサーバのデータ送信は、1つのアンカーあたり1秒間におおむね4回づつ行われ、1回あたりの送信データ量は(a)で最大約250Kbyte、(c)で最大約400Kbyteとなった。

(a)では最大で4個のアンカーが、(c)では、最大7個のアンカーが、それぞれ検出されており、アンカー毎にデータ量が異なっている。これは、LiDARセンサーが検出できる距離内にあるポリゴンメッシュの数が、アンカー毎に異なるためである。また、および、現実世界の地形や物体の形状の複雑さによってもデータ量は異なり、(c)の樹木のように形状が複雑な場合にデータ量は大きくなる。

ARアプリケーションとしては、積雪オブジェクトが表示されるまでの時間的な遅延はあるものの、使用したiPhoneのAR表示性能が高いこともあり、デバイスの向きの変化に積雪オブジェクトが追従して違和感なく画面上に表示された。エッジサーバへのオフローディング処理が実行されているにもかかわらず、画面上の処理が停止する等のユーザ体験の品質低下は見られなかった。

4.3 考察

本システムは実装途上にあり、いくつか未実装な点がある。まず、風の影響や冠雪を厳密に再現した積雪オブジェクトを生成するには、形状や雪質を考慮した積雪のシミュレーションが必要となる。そうしたシミュレーションを含む調整処理の検討・実装は今後の課題である。積雪形状のモデリングに関する既存研究 [8] 等の適用も検討したい。

また、現実空間の上部に物体があり、積雪しない状態となる位置があることに対応できていない。当初はLiDARで検出できていなかった上部に物体が検出された場合、それまでに生成された積雪オブジェクトが実際には積雪しない位置となることがあるため、そうした場合に非表示とする処理等が必要となる。

また、積雪オブジェクトは三角形の集合によるポリゴンメッシュで表現されるが、提案システムの積雪オブジェクト生成アルゴリズムでは、物体の形状とLiDARによる点群の検出状況によっては、直線であるべき側面部分に三角形が表示されるなど、不自然に見えることがある。より精緻な積雪状況を表示するにはポリゴンメッシュを構成する三角形の大きさを小さくした上で、数を増加させれば良いが、データ量・計算量も増大してしまう。表示処理の性能はモバイルデバイスごとに異なると考えられ、メモリ量や処理性能に応じて精度を調整するなどの工夫が必要となる可能性がある。また、iOSが有するARKitのように、高度なAR表示機能がないモバイルデバイスでは、表示映像をエッジサーバ側で行うなど、処理の分担を変化させた方が良い場合もあると考えられる。

本実装では、アンカー毎のデータが得られる度に、エッジサーバへデータを送信し、積雪オブジェクトを生成し、ARKitを用いて表示している。しかし、LiDARを用いて検出される点群データは、短時間で大きく変化することはないため、検出される点群データに基づく積雪オブジェクトの生成処理を毎回行わずとも、画角内の積雪オブジェクトのAR表示を継続することは可能と考えられる。どの程度の頻度で点群データの送信や積雪オブジェクトの生成を行うべきかは、アンカー内の検出点群データ数、モバイルデバイスの移動速度、向きの変化の度合いによって異なると考えられる。適切な頻度での積雪オブジェクト生成により冗長な通信や処理を削減する方法の検討は今後の課題である。

また、本実装では、LiDARにより点群データが検出された範囲に対応する積雪オブジェクトを生成・表示するた

め、図3の表示における上部の歩道部分など、LiDARによる点群データ検出可能距離を超えた遠方には、積雪オブジェクトがレンダリングされない。このような画角内の位置に積雪オブジェクトを表示するには、ユーザが移動してLiDARの検出範囲を広げ、点群データを保持するアンカーを増やす必要がある。このとき、例えば過去に同ユーザや他ユーザが同じ位置で本システムを利用しており、既にエッジサーバ上に同位置の空間情報が存在すれば、ユーザが移動せずとも遠方の積雪オブジェクトを生成し、AR表示することは可能となる。観光地などでは同じ位置で複数のユーザが積雪状況を確認する場合があると考えられ、そのような空間情報の共有が有効となる可能性がある。ただし、現状では点群データとして得られる座標系は、先に述べた通り、ユーザがアプリケーションを起動した時のデバイス位置・向きからの相対的な座標を用いているため、空間情報の共有座標へのマッピングや調整が必要となる。上記のような現実世界の空間情報の座標共有が可能となれば、現実世界とリンクしたメタバース等のさまざまな応用に適用できる可能性もある。

5. おわりに

本稿では、我々が設計開発を進めている空間情報処理のオフローディングによるリアルタイムAR積雪可視化システムの実現方法について述べた。初期実装により、エッジサーバとオフローディング処理で連携動作する提案システムが、違和感なく積雪オブジェクトをモバイルデバイス上にAR表示できることを確認した。

4.3節で述べた通り、現在の設計および実装にはいくつか課題がある。特に、冠雪を含む降雪オブジェクトの形状調整を、積雪シミュレーションによって行うモジュールの実装が第一の課題として挙げられる。よりリアルな積雪状態を再現可能とするため、積雪シミュレーションにおいては、風向や気温による積雪状態の変化等の考慮も検討したい。

また、ユーザのAR体験の品質を変化させない範囲ですでに処理を行った部分の冗長な処理の省略を行う方法や、他のユーザによる同じ場所のセンシング結果を再利用するなどの工夫について、今後さらに検討・設計を進める。

謝辞

本研究の一部はJSPS科研費22K12009, 21H03429, 18K11316, 放送文化基金, 「北陸地域の活性化」に関する研究助成事業, G-7奨学財団研究開発助成事業による成果である。

参考文献

[1] 今枝卓也, 高汐一紀, 徳田英幸: uMegane: AR技術を用いたセンサ情報可視化システム, 情報処理学会研究報告, Vol. 2008-UBI-019, No. 66, pp. 39-44 (2008).

- [2] ウェザーニュース: AR お天気シミュレーター, 株式会社ウェザーニュース(オンライン), 入手先 (<https://wxar.weathernews.jp/app/about/download.html>) (参照 2022-05-22).
- [3] Taleb, T., Samdanis, K., Mada, B., Flinck, H., Dutta, S. and Sabella, D.: On Multi-access Edge Computing: A survey of the emerging 5G network edge cloud architecture and orchestration, *IEEE Communications Surveys & Tutorials*, Vol. 19, No. 3, pp. 1657-1681 (2017).
- [4] Tanaka, H., Yoshida, M., Mori, K. and Takahashi, N.: Multi-access Edge Computing: A Survey, *Journal of Information Processing*, Vol. 26, pp. 87-97 (2018).
- [5] 平島寛行: 雪国の情報革命 ~雪おろシグナル~, 防災科研ニュース, No. 204, pp. 4-5 (2019).
- [6] 伊豫部勉, 河島克久: 準リアルタイム積分分布監視システムの開発, 日本雪工学会論文集, Vol. 36, No. 1, pp. 1-13 (2020).
- [7] 高瀬恵次, 小倉 晃, 藤原洋一, 丸山利輔: 積雪深の再現を目的としたモデルの構築と検証, 水文・水資源学会誌, Vol. 29, No. 2, pp. 107-115 (オンライン), DOI: 10.3178/jjshwr.29.107 (2016).
- [8] 床井浩平, 森木大樹: 積雪形状のリアルタイムモデリング, 情報処理学会論文誌, Vol. 47, No. 5, pp. 1558-1565 (2006).
- [9] Apple: iPhone, Apple Inc. (online), available from (<https://www.apple.com/jp/iphone/>) (accessed 2022-05-22).
- [10] Apple: ARKit, Apple Inc. (online), available from (<https://developer.apple.com/jp/documentation/arkit/>) (accessed 2022-05-22).