

拡張現実感技術に基づくメタ AR 空間構築のための マッピング支援機構の開発

安江 洗希^{†1,a)} 菊地 真人^{†1,b)} 大冢 忠親^{†1,c)}

概要：現実空間をデジタル空間上に再現するための 3 次元空間構築技術は、様々な場面での活用が期待されている。本研究の目的は、現実空間から疑似的に構築された仮想の 3 次元空間を用いて、現実空間と仮想空間の間でのインタラクションを実現することである。そのための空間をメタ AR 空間と呼ぶ。メタ AR 空間の構築には、ARKit から取得した 3D メッシュモデルの各ポリゴンにテクスチャを割り当てるメタ AR 空間構築処理を実現する必要がある。ここで、ユーザによるメタ AR 空間構築処理のための作業をマッピングと呼ぶ。マッピングにおいてユーザは、LiDAR スキャナや RGB-D カメラを用いて現実空間を取り込む。ここでの課題として、マッピング中のユーザは、現実空間が過不足無くマッピングされているかを知ることができないという点が挙げられる。すなわち、ユーザにとってマッピングを終える適切なタイミングがわからない。その結果、マッピングが不十分であったり、必要以上に丁寧なマッピングによる作業時間の長時間化という問題が生じる。本研究では、マッピングされていない場所をリアルタイムに可視化することが可能なマッピング支援機構を実現することでこの問題の解決を目指す。本稿では、メタ AR 空間の構築手法およびマッピング支援機構について述べる。

1. はじめに

現実空間をデジタルな空間上に再現する 3 次元空間構築技術は、様々な場面での活用が期待され、近年注目が集まっている。視覚的な情報の再現は、現地に赴かずとも、その場所にいるかのような体験の提供を可能とする。また、デジタル空間上では物理的制約を受けないため、現実では行うことが難しい作業をシミュレーションによって体験することができ、トレーニングなどでの活用が可能となる。再現する 3 次元空間は、再現性が高いことが望まれる。しかし、再現性の高い 3 次元空間を構築するには、高精度なセンサや高速なコンピュータが必要である。

一方、汎用のスマートフォンやタブレット端末にも LiDAR スキャナ等の距離センサが搭載されるようになり、高度な拡張現実感技術 (AR) が実現されている。そこで本研究では、Apple 社の開発した ARKit^{*1}等の汎用機器に搭載された AR 機能による環境情報取得機能を利用することで、汎用的なデバイスのみで 3 次元空間構築を実現することを目指す。さらに、その場にいなくても AR 空間をメタ

的に操作可能にするための仮想空間であるメタ AR 空間を開発している。メタ AR 空間により、遠隔地からの誘導支援のように、AR 技術を用いた新たな遠隔作業支援の実現を可能にする [1]。

本研究では、汎用スマートフォンなどを用いて構築したメタ AR 空間の再現性の向上を目指している。例えば、本研究の先行研究では、汎用スマートフォン (以降、単にデバイスと呼ぶ) を用いて、現実空間を 3 次元点群として構築するシステムを開発した [2]。ここでは、LiDAR スキャナが搭載されたデバイスにより、現実空間を 3 次元点群データとして取得し、さらに点群に彩色することでメタ AR 空間を構築した。しかし、LiDAR スキャナによって得られる点群の密度は低いため、空間の大まかな形状等の表現は可能であるが、例えば、ポスターの文字のような細かい情報を表現できなかった。そこで、本研究では ARKit により取得可能な 3 次元メッシュモデル (3D メッシュモデル) を構成するポリゴンにテクスチャを割り当てることによる、メタ AR 空間構築処理技術を開発する。本稿では、ユーザが、メタ AR 空間構築に必要な情報を本システムを用いて収集する作業のことをマッピングと呼ぶ。

ポリゴンへのテクスチャの割り当てをリアルタイムで処理することができないため、ユーザは、現実空間が過不足無くマッピングされているかを、マッピング中に知ることが

^{†1} 現在、名古屋工業大学

Presently with Nagoya Institute of Technology

a) ankou@ozlab.org

b) kikuchi@nitech.ac.jp

c) ozono@nitech.ac.jp

*1 <https://developer.apple.com/jp/augmented-reality/arkit/>

できない。その結果、マッピングされていない場所が残ったり、マッピングされていない場所を減らすための丁寧なマッピングによる作業時間の長時間化が問題となる。不十分なマッピングでは、テクスチャがない場所が残り、再現性の低いメタ AR 空間になる。本研究では、本問題を解決するためにマッピング支援機構の実現を目指す。マッピング支援機構は、マッピング時に、メタ AR 空間構築処理のための適切な情報が取得できているか否かを可視化する。これにより、ユーザはマッピングすべき場所の把握が可能になり、再現性の高いメタ AR 空間の構築が期待される。

本稿の以降の構成について述べる。2章では、メタ AR 空間やメタ AR 空間を構築するためのマッピングおよびマッピング支援について述べる。3章では、メタ AR 空間構築処理の実装について説明し、マッピング支援の実現に向けて必要な処理について述べる。4章では、マッピング支援機構について述べ、5章で本研究の関連研究を紹介する。最後に6章で本稿についてまとめる。

2. メタ AR 空間と構築支援

2.1 AR 空間とメタ AR 空間

本節ではメタ AR 空間に関して、図1を用いて説明する。AR 空間とは、現実空間に対して何らかのデバイスを通して視認可能な空間のことで、デジタル情報（AR オブジェクト）を自由に付加することができる。また、AR を利用することで、現実空間の形状を取得することが可能である。AR では、現実空間の形状に関する情報を利用して、仮想のオブジェクトを、現実のオブジェクトであるかのように表示することを可能にする。ただし、AR 空間は、現実空間に仮想の AR オブジェクトを重畳表示した空間であるため、AR 空間内の AR オブジェクトを操作するには、ユーザが現地にいる必要がある。すなわち、操作可能な AR 空間は、ユーザの居場所に依存する。

本研究におけるメタ AR 空間は、操作可能な AR 空間がユーザの居場所に依存するという制約を取り除いた AR 空間である。メタ AR 空間では、AR 空間における現実空間の 3D モデルを仮想空間内に再現する。AR 空間とメタ AR 空間の間では双方向インタラクションを可能にする。メタ AR 空間を操作することは、AR 空間を操作することに相当し、逆に、AR 空間を操作することは、メタ AR 空間を操作することに相当する。これにより、ユーザの居場所に制約されない AR 空間のメタ的な操作を可能にする。

2.2 メタ AR 空間構築におけるマッピング

本稿では、メタ AR 空間を構築するために現実空間の形状や色等の情報を収集する作業をマッピングと呼ぶ。例えば、ユーザは、AR 空間として構築したい部屋や建物等を、デバイスのカメラや LiDAR スキャナを用いて撮影する。ここで、AR 空間の構築と同時並行的にメタ AR 空間も

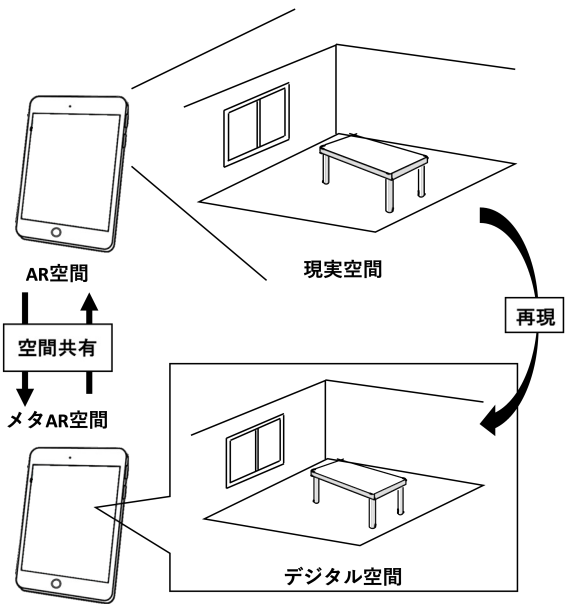


図1: AR 空間とメタ AR 空間

構築することで、共通の座標系を持つ AR 空間とメタ AR 空間を構築する。構築時間は、AR 空間よりもメタ AR 空間の方が長い。理由としては、AR 空間の構築は、空間における形状がわかればよい。一方、メタ AR 空間の構築には、空間の形状に加えテクスチャの情報が必要である。

最も効率的なマッピングとは、マッピングした部分が重複したりマッピングしていない部分がない、過不足のないマッピングをすることである。特にマッピングが不足すると、メタ AR 空間の再現性に悪影響がある。前述したように、メタ AR 空間の構築は遅いため、ユーザは、マッピング中に過不足を確認することができない。すなわち、メタ AR 空間を構築することなしに、何らかの方法でユーザにマッピング不足を伝えることが必要である。

2.3 マッピング支援

マッピング支援とは、ユーザのマッピング作業に対して不足のないマッピングが行えるように支援することである。具体的には、リアルタイムにマッピング状態を管理することで、マッピング中のユーザにマッピングすべき場所を伝える。マッピングすべき場所とは、メタ AR 空間を構築するための情報が不足している位置のことである。特に、形状情報よりもテクスチャの不足が顕著な問題となるので、その点を支援する。メタ AR 空間の構築情報が不足していることをユーザに示すことで、再現性の高いメタ AR 空間の構築を支援する。また、既にマッピング済みの場所を重複してマッピングする無駄を省くことができれば、効率的なマッピングにも繋がる。

3. メタ AR 空間の構築

本章では、まず 3D メッシュモデルにテクスチャを割り当

てるメタ AR 空間構築処理について説明する。次に、マッピング支援が必要となる処理についてまとめる。

3.1 ARKit に基づくメタ AR 空間の構築

メタ AR 空間は、ARKit を用いて取得可能な 3D メッシュモデルにテクスチャを割り当てるメタ AR 空間構築処理によって構築する。ここでの 3D メッシュモデルとは、ポリゴンの集合を指す。この 3D メッシュモデルは、ARKit によって、LiDAR スキャナが搭載されたデバイスで取得可能な深度情報を基に作成される。3D メッシュモデルの情報は、ARMeshAnchor に格納されている。ARMeshAnchor とは、ARKit が検出した 3D メッシュモデルによって構成された物理オブジェクトである。マッピングを通して ARKit は ARMeshAnchor を一定の範囲ごとに切り替えて、新しい ARMeshAnchor 作成する。そのため、スキャンする空間の大きさに比例して 3D メッシュモデルを構成する ARMeshAnchor の数は増える。ARMeshAnchor の内部情報としては以下ようになる。

- geometry：3D メッシュモデルの形状情報
 - vertices：頂点の座標
 - faces：3 角ポリゴンの頂点情報
 - normals：法線情報
 - classification：ポリゴンのクラス分類（8 クラス）
- transform：ワールド座標系に変換する行列
- identifier：アンカー固有の ID

geometry（ジオメトリ）とは、オブジェクトの形状を定義するための頂点やポリゴンで構成されたデータのことである。メタ AR 空間構築処理では、ARMeshAnchor のジオメトリ情報を基に行う。図 2 に、メタ AR 空間構築処理の概要を示す。図 2 には、マッピングして取得したある瞬間の RGB 画像とその位置の ARMeshAnchor を基にデジタル空間上に 3D メッシュモデルを表示している。ARMeshAnchor による 3D メッシュモデルの可視化は、RealityKit^{*2}を用いることで実現できる。しかし、RealityKit では、表示する 3D メッシュモデルの色やテクスチャを操作する機能が提供されていない。そこで本研究では、SceneKit^{*3}を用いて、3D メッシュモデルの可視化や内部情報の操作といった処理を行っている。SceneKit におけるジオメトリの作成には、SCNGeometry クラスを用いる。具体的には、ARMeshAnchor オブジェクトが表す頂点、面、そして法線情報から SCNGeometry オブジェクトを生成する。また、この SCNGeometry に与える頂点に図 2 のように UV 値を与えることで、与えた UV 値に従ってテクスチャをポリゴンに割り当てた 3D モデルの構築が可能となる。

本研究のメタ AR 空間構築処理では、テクスチャの無い

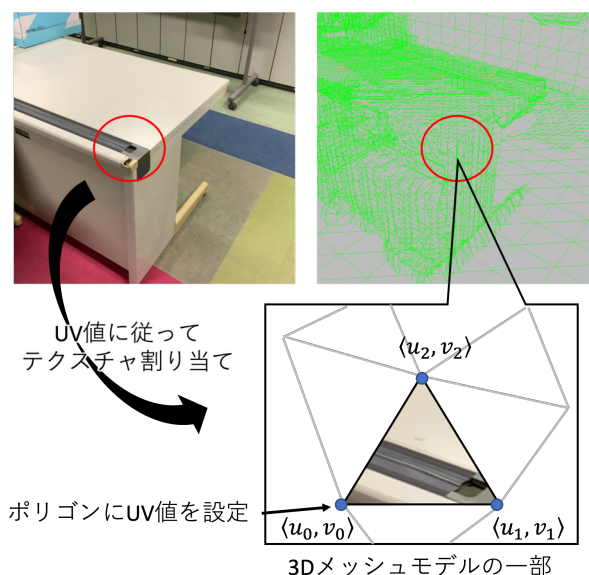


図 2: メタ AR 空間構築処理の概要

3D メッシュモデルの各ポリゴンに、マッピング時に取得した複数の RGB 画像等のデータを用いて、テクスチャを割り当てる。ここで、RGB 画像は、後述する 2 次元のスクリーン座標系上にあり、3D メッシュモデルは 3 次元空間上にあるため、3D メッシュモデルの各ポリゴンに割り当てる画像を決定するには、何らかの処理が必要である。具体的には、次の 4 ステップから構成される。最初にステップ 1 では、ポリゴンの各頂点の座標をスクリーン座標系に変換する。次にステップ 2 では、スクリーン座標に変換したポリゴンの奥行きを判定する。その後ステップ 3 では、割り当てたテクスチャが歪まないように 3D メッシュモデルのジオメトリ情報を拡張する。最後にステップ 4 では、条件を満たすポリゴンの各頂点の UV 値を求める。以降、ステップ 1 の座標変換では、ポリゴンの 3 次元座標をデバイス画面の 2 次元座標系であるスクリーン座標系に変換する。以降、3.1.1 ~ 3.1.3 で各ステップについて説明する。

3.1.1 ポリゴン座標の変換と奥行き判定

前述したように、ポリゴンへのテクスチャの割り当ては、2 次元の画像を 3 次元メッシュモデル内の平面として配置したときに、3D メッシュモデルをその平面に射影すればよい。当然、3D メッシュモデルにおける最前面のポリゴン、すなわち、その 2 次元平面から見て可視であるポリゴンのみを考慮する必要がある。2 次元平面からのレイトラッキングにより、その画像に該当するポリゴンを特定することが可能であるが、レイトラッキングは計算コストが高い。よって、本研究では、RGB-D 画像の深度情報 (DepthMap) を利用した簡易的な前後判定を行う。図 3 に DepthMap を用いた奥行き判定の概要を示す。DepthMap には、取得した瞬間のスクリーン上に映る物体の、表面に関する深度情報が含まれており、ピクセル毎に 3 次元座標に変換することができる。判定方法としては、ポリゴンの

^{*2} <https://developer.apple.com/jp/augmented-reality/realitykit/>

^{*3} <https://developer.apple.com/documentation/scenikit/>

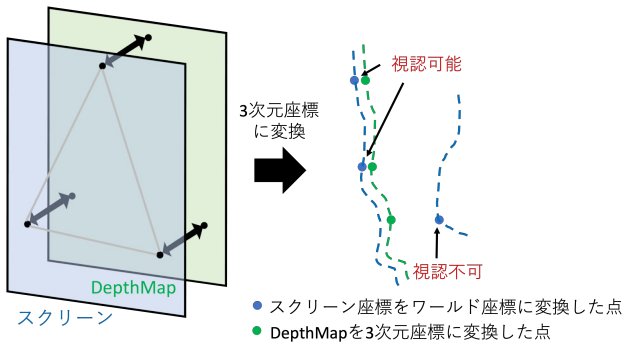


図 3: DepthMap による奥行き判定

各頂点のスクリーン座標がスクリーン内に含まれている場合に、スクリーン座標をワールド座標系に変換した際の3次元座標と、DepthMapのスクリーン座標部分を3次元座標に変換した際の座標の間の距離を比較する。この距離が一定距離内にあれば最も手前側にあるポリゴンとする。

3.1.2 3D メッシュモデルのジオメトリ拡張

本システムでは、UV マッピングに使用するテクスチャとして、マッピングで取得した複数の RGB 画像を1枚の RGB 画像として使用する。しかし、ARMeshAnchor で作成される 3D メッシュモデルのジオメトリは、隣り合うポリゴンと頂点を共有しているため、ポリゴン間で別々の RGB 画像の割り当てが発生した場合に片方のポリゴンのいずれかの頂点に対応する UV 値が更新されてしまい、テクスチャの歪みが発生する。そのため、ポリゴン間で独立した頂点を持つように 3D メッシュモデルのジオメトリを拡張する。図 4 に 3D メッシュモデルのジオメトリ拡張の例を示す。図 4 には、2つのポリゴンで構成された 3D メッシュモデル、各ポリゴンを構成する頂点情報で構成された faces、各頂点の 3次元座標 $\langle x, y, z \rangle$ で構成された vertices の 3つが存在する。この faces と vertices から、ポリゴンの頂点が独立するように拡張した ReFaces と ReVertices を作成する。ReVertices は、ポリゴン間で頂点を独立させるためポリゴンの数 \times 3 個分の頂点座標を持つようにする。この際に、変更前の faces のポリゴンを構成する頂点を複製した頂点を用いる。また、ReFaces は、これらの複製された頂点を追加する。すなわち、変更前後のポリゴンにおいて、同じ座標の頂点が重複することになる。図 4 には示していないが ARMeshAnchor に含まれる normals も vertices と同様の操作を行い、本システムでは ReNormals として作成し利用する。

3.1.3 UV 値の計算

図 5 にメタ AR 空間構築処理で使用するテクスチャの例を示す。マッピングによって取得した RGB 画像を並べて、テクスチャを作成している。取得した RGB 画像の横の長さを W 、縦の長さを H とする。また、テクスチャの中で、横に並べた RGB 画像の枚数を $count_w$ 、縦の枚数を

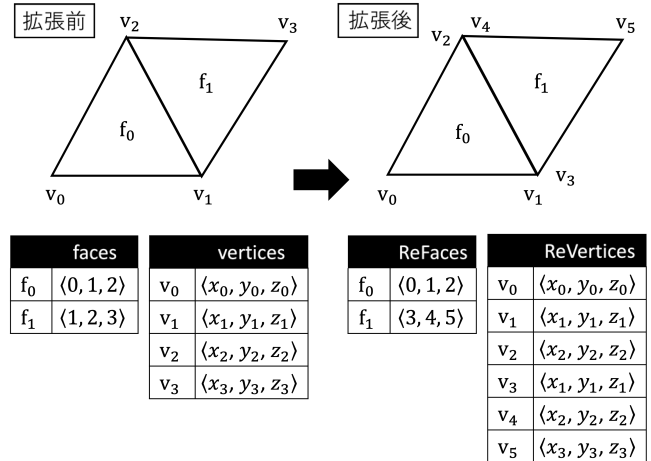


図 4: 3D メッシュモデルのジオメトリ拡張の例

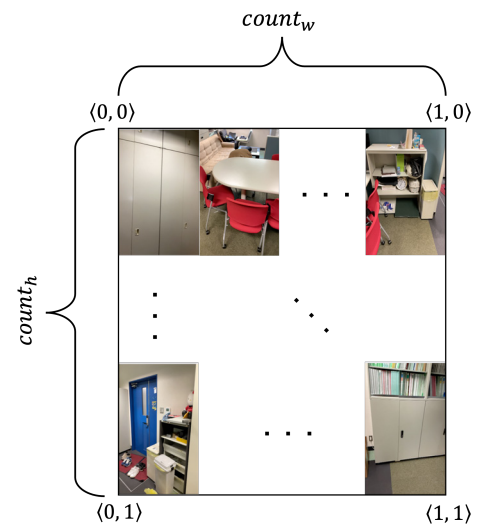


図 5: マッピングに使用するテクスチャの例

$count_h$ とする。この時、 i 番目の RGB 画像と対応するスクリーン座標 (p_x, p_y) についてポリゴンの UV 値は式

$$u = \frac{p_x}{W \cdot count_w} + \frac{i \bmod count_w}{count_w}$$

$$v = 1 - \left(\frac{p_y}{H \cdot count_h} + \frac{i}{count_w \cdot count_h} \right)$$

で求まる。V 値の計算において、スクリーン座標系（左下が原点）と UV 座標系（左上が原点）で上下方向が逆であることを考慮する必要がある。すなわち、スクリーン座標から UV 座標へと変換する際に V 値に -1 を乗じている。

3.2 マッピング可能性判定問題

本稿では、取得済みの画像に関して、3D メッシュ中のポリゴンのテクスチャが決められるかどうかを決定する問題を、マッピング可能性判定問題と呼ぶ。前述のテクスチャ割り当て処理を実行すれば、マッピング可能性判定問題を解くことができる。本研究では、テクスチャ割り当て処理よりも高速にマッピング可能性判定問題を解くためのアプ

Algorithm 1 Find reasonable faces for RGB-D images.

```
1: function FINDFACES( $\{\pi_1, \pi_2, \dots, \pi_N\}, \{f_1, f_2, \dots, f_M\}$ )
2:   result  $\leftarrow$  []
3:   for  $\pi \in \{\pi_1, \pi_2, \dots, \pi_N\}$  do
4:      $A \leftarrow$  projectionMatrix( $\pi$ )
5:      $D \leftarrow$  depthMap( $\pi$ )
6:     for  $f \in \{f_1, f_2, \dots, f_M\}$  do
7:        $f' = A \cdot f$  ▷ ( $f$  を  $\pi$  に射影)
8:       if  $f'$  is on  $\pi$  then
9:         score  $\leftarrow$  verifyDepthConstraint( $f, D$ )
10:        if score  $\leq$  0.04 then
11:          result.append( $f$ )
12:        end if
13:      end if
14:    end for
15:  end for
16:  return result
17: end function
```

ローチを検討している。具体的には、マッピング中にリアルタイムにマッピング可能性判定問題を解くことを狙っている。基本的なアイデアとしては、3.1.1 をマッピング可能性判定問題を解くために最適化すればよい。

Algorithm 1 にマッピング可能性判定マッピング支援でのアルゴリズムを示す。Algorithm 1 は、取得済みの全画像が、何れのポリゴンにテクスチャ割り当てできるのかを、探索するアルゴリズムである。入力は、RGB-D 画像の集合 $\{\pi_1, \pi_2, \dots, \pi_N\}$ およびポリゴンの集合 $\{f_1, f_2, \dots, f_M\}$ である。出力は、テクスチャ割り当てが可能なポリゴンの集合である。Algorithm 1 は、2 行目でテクスチャ割り当てが可能なポリゴンのリスト $result$ を初期化する。3 行目で 1 枚の RGB-D 画像 π を選択する。 π は、3D メッシュモデル内での有限な平面である。4 行目および 5 行目では、それぞれ π の射影行列 A および深度情報 D を取得する。 A は、 $f \in \{f_1, f_2, \dots, f_M\}$ を平面 π に射影するための変換行列である。 D は、RGB-D 画像に含まれる深度情報であり、 π 上の任意の点から法線方向に存在する最も近い面との距離を表している。これを、 π と f の距離推定に利用する。6, 7 行目で A を用いて f を π に射影し、 f' を得る。 π は有限な平面であるから、8 行目で f' が π 上にあるかを調べる必要がある。9 行目では、 π と f の距離が D と無矛盾かどうかの度合い $score$ を得る。 D の測定誤差を考慮して、10 行目で $score \leq 0.04$ ならば、 f をマッピング可能であるとして受理する。その後、 f を $result$ に追加する。最後に、16 行目で $result$ を返す。

4. マッピング支援機構

マッピング支援機構は、ユーザが行うメタ AR 空間構築のためのマッピングに対して、適切なマッピングが行えるような支援を行う機構である。マッピング支援機構を用いたマッピングの実行イメージを図 6 に示す。本システムで

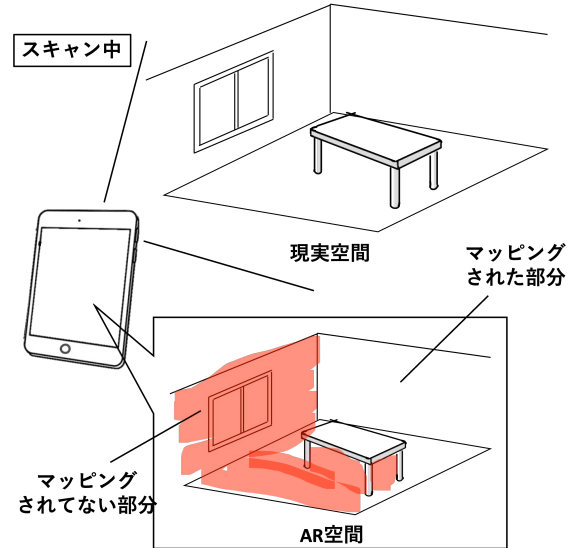


図 6: マッピングの実行イメージ

は、マッピングの状態を AR 空間上に重畳表示し、マッピングされている部分とされていない部分を直感的に判断しやすいシステムとする。マッピングされているとは、メタ AR 空間構築処理を行うのに十分なパラメータの取得ができていることを指す。また、マッピングされていないとは、十分なパラメータの取得ができていないことを指す。マッピング支援機構は、マッピング開始時に AR 空間全体に赤色の重畳表示を行う。そして、マッピングを通して、3D メッシュモデル中のポリゴンに対してテクスチャの割り当てが可能となった部分から赤色の重畳表示を取り除いていき、現実空間をそのまま視認できるようにする。これによってユーザは、直感的にマッピングすべき位置がわかるようになり、効率のよいマッピングが可能となる。

4.1 システム構成

システム構成図を図 7 に示す。本システムは、メッシュ情報管理機構、パラメータ管理機構、マッピング支援機構の 3 つの部分から構成されている。メッシュ情報管理機構は、ARKit によって作成される ARMeshAnchor を管理する機構である。ARMeshAnchor は、マッピングを通して取得した深度情報などから常に最適な 3D メッシュモデル情報に更新される。ARMeshAnchor の更新が発生した際に、その ARMeshAnchor をマッピング支援機構に送る。パラメータ管理機構は、メタ AR 空間の構築に必要な RGB 画像や DepthMap などのパラメータを取得し、管理する機構である。マッピング支援機構に ARMeshAnchor が送られた際に、その時点で保存したパラメータを全てマッピング支援機構に送る。マッピング支援機構は、送られてきた ARMeshAnchor とパラメータを用いて Algorithm 1 を実行し、テクスチャ割り当てが可能である部分を求めて、その処理結果を AR 空間上に反映する。

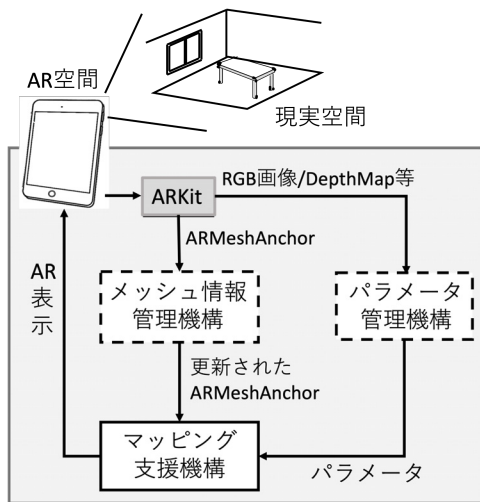


図 7: システム構成図

4.2 リアルタイム性の実現

Algorithm 1 は、マッピング中に ARMeshAnchor のポリゴン情報にアクセスし、テクスチャの割り当てが可能かを判定する処理（マッピング可能性判定処理）である。マッピング支援機構の実現において、Algorithm 1 をリアルタイムに処理する必要がある。素朴な実装では、1 秒以上を要するため、リアルタイム性を実現するには高速化が必要である。また、リモートのサーバ等を用いて計算する方法も考えられるが、送受信するデータサイズが懸念される本システムでは、iOS の GPGPU 計算フレームワークである Metal^{*4}を用いて、デバイス上での処理の高速化を目指す。

Metal とは、GPU への低レベルなアクセスを可能にし、グラフィックの描画や演算処理で高いパフォーマンスを発揮することを可能とするフレームワークである。本研究では、Metal が汎用的な並列計算に利用可能である点に着目した。3D メッシュモデルを構成する各ポリゴンに対する処理は他のポリゴンに対する処理と独立しているため、マッピング可能性判定処理は、容易に並列計算が可能である。

5. 関連研究

現実空間の環境情報をデジタル空間上に 3 次元構築し利用する研究は多数存在する。Aoki ら [3] は、美術館の展示とすることが難しいコレクション（崩れかけの建物、昔の風景など）をデジタル空間上に再現することで、ユーザが自由に鑑賞できるようなシステムを開発した。ユーザは、物理的な制約を無視できるため、デジタルコンテンツを自由に操作することが可能となる。これによって展示されたコレクションに対してより深く理解することが期待できる。Rajeev ら [4] は、GoogleEarth 画像と SAR データ（衛星による地球表面のレーダースキャンデータ）を用いてデジタル空間上に 3D マップを作成し、この 3D マップを集めた 3D 景観データベースを構築した。このデータベースを

用いて、未知の環境でのナビゲーション支援やミッションに対する支援の有効性を検証している。これらは主に視覚的な情報をデジタル空間上に再現して利用した例である。

Aussama ら [5] は、デジタル空間を用いたシミュレーションによる、技術的スキルと非技術的スキルの習得を支援するようなトレーニング方法を提案している。現実空間上では危険が伴い行うことが難しいようなトレーニングもデジタル空間上では安全に行うことができるため、スキルの習得に役立てることが可能となる。Stotko ら [6] は、テレプレゼンスのための大規模な 3 次元空間構築手法を開発している。これは、遠隔でのコラボレーションやコンサルティングに適したシステムとなっており、ユーザは、スマートフォンや Kinect のような汎用的なデバイスで RGB-D 画像を取得し、リアルタイムに 3 次元構築をすることで高い没入感を得ることができるようなシステムとなっている。

6. おわりに

本稿では、3D メッシュモデルにテクスチャを割り当てる処理であるメタ AR 空間構築処理について説明した。ここでは、メタ AR 空間構築処理に必要な情報収集作業であるマッピングにおける課題について述べた。本研究では、マッピングが不完全な部分を可視化することで、作業中のユーザを支援するマッピング支援機構について示した。本システムが実現されることで、効率のよいマッピングが可能になることが期待される。しかし、リアルタイムなマッピング可能性判定処理の実現には課題が残されており、今後の課題である。

謝辞 本研究の一部は JSPS 科研費 JP19K12097, JP19K12266. の助成を受けたものです。

参考文献

- [1] 安江洗希, 大園忠親, 新谷虎松, “拡張現実感技術によるナビゲーション支援のための AR オブジェクト共有機構の試作”, 信学技報, Vol.120, No.344, AI2020-19, pp.30-35, 2021.
- [2] 安江洗希, 菊地真人, 大園忠親, “メタ AR 空間構築のための汎用タブレット端末で実行可能な 3 次元点群差分検出手法について” SMASH21 Summer Symposium, Vol.2021-ICS-204, No.17, pp.1-4, 2021.
- [3] A. Taiju, et al., “Basic Study on Connecting AR and VR for Digital Exhibition with Mobile Devices”, 19th International Conference on Human-Computer Interaction, pp.101-112, 2017.
- [4] S. Rajeev, et al., “3D Navigational Insight using AR Technology”, IEEE International Symposium on Technologies for Homeland Security, pp.1-4, 2019.
- [5] N. Aussama, et al., “Virtual reality (VR) as a simulation modality for technical skills acquisition”, Annals of Medicine and Surgery, No.71, pp.1-7, 2021.
- [6] P. Stotko, et al., “SLAMCast: Large-Scale, Real-Time 3D Reconstruction and Streaming for Immersive Multi-Client Live Telepresence”, IEEE Transactions on Visualization and Computer Graphics, pp.2102-2112, 2019.

*4 <https://developer.apple.com/jp/metal/>