

エネルギー科学研究科  
エネルギー社会・環境科学専攻修士論文

題目： 実世界指向プログラミングを用いた  
AR訓練環境構築システムの開発

指導教員： 下田 宏 教授

氏名： 田村 太一

提出年月日： 令和2年2月14日(金)

## 論文要旨

題目：実世界指向プログラミングを用いた AR 訓練環境構築システムの開発

下田宏研究室，田村 太一

### 要旨：

昨今では原子力関連施設などの重要施設における安全性の見直しが進んでおり、中でも防災教育の充実などのソフト面の対策が注目されている。ソフト面の対策の1つとして、現場で実施される防災訓練や不法事案対策訓練が挙げられるが、これらには、参加できる人および体験できる訓練シナリオが限られるという問題点がある。そこで、それらの問題点を解決できる拡張現実感を用いた訓練（AR 訓練）が注目されている。

しかし、AR 訓練環境の構築には専門知識が必要であるため、専門企業に外注されることが多いが、その企業は現場の状況や訓練内容に精通しているわけではない。そのため、作成できる訓練シナリオの種類が限られてしまう、現場で求められる AR 訓練環境と乖離してしまうなどの可能性がある。また、現場の訓練環境を直接観察しながら AR 訓練環境を構築することは難しい。上記のように、現状、AR 訓練環境の構築にあたっては、専門知識が必要であること、現場での訓練シナリオ作成が困難であることが課題となっている。

そこで本研究では、プログラミングなどの高度な情報系のスキルを持っていない人でも容易に、現場を直接観察しながら AR 訓練環境を構築できるシステム、AR 訓練環境構築システム（ARTM）を開発することを目的とする。本研究ではプログラミングに関する専門知識を持っていない人でも容易に理解及び操作でき、かつ現場で扱えるように、新しい実世界指向プログラミング言語、「プログラムブロック」を開発した。プログラムブロック設計の際には、重要施設で実施されている主な訓練を参考にし、現場で求められている訓練シナリオを実現するのに適したプログラムブロックの機能を選定した。また、プログラムブロックの形状をユーザが直感的に理解できるように設計した。

本研究では、防災訓練や不法事案対策訓練を定期的に行っている重要施設の職員に ARTM を使用してもらい、容易に理解および操作が可能か、ARTM で現場で求められている水準の複雑さの訓練シナリオを作成できるかを評価した。本評価により、従来の AR 訓練環境の構築方法と比べると、ARTM を用いることにより大幅に構築が容易になったことが分かった。また、ARTM で作成できる訓練シナリオの自由度は現状では十分高いわけではないが、複数の追加機能によって現場で求められている AR 訓練を作成できるようになると分かった。

# 目次

第 1 章 序論	1
第 2 章 研究の背景と目的	3
2.1 AR 訓練の現状と問題点	3
2.2 AR 訓練環境構築に関する既往研究	5
2.3 研究の目的と意義	6
第 3 章 AR 訓練環境構築システム (ARTM) の開発	7
3.1 システムの目標	7
3.2 ARTM の概略設計	7
3.3 プログラムブロックの設計	11
3.3.1 プログラムブロックで実現する訓練要素の検討	11
3.3.2 プログラムブロックの機能の検討	19
3.3.3 プログラムブロックの形状の検討	24
3.4 訓練環境構築サブシステムの設計	31
3.4.1 トラッキング方法と表示デバイスの検討	31
3.4.2 訓練環境構築サブシステムの詳細設計	35
3.5 訓練環境体験サブシステムの設計	40
3.5.1 トラッキング方法と表示デバイスの検討	40
3.5.2 訓練環境体験サブシステムの詳細設計	41
3.6 ARTM の実装	44
3.6.1 訓練環境構築サブシステムの実装	44
3.6.2 訓練環境体験サブシステムの実装	59
3.6.3 ARTM の動作例	67
第 4 章 システムの評価	70
4.1 評価の目的と概要	70
4.2 評価方法	71

4.2.1	評価項目	71
4.2.2	課題内容	72
4.2.3	評価環境	73
4.2.4	評価手順	74
4.2.5	評価アンケート	77
4.2.6	評価者	79
4.3	評価結果	79
4.4	結果の考察	84
4.4.1	ARTMの容易さに関する評価結果の考察	84
4.4.2	訓練シナリオの自由度に関する評価結果の考察	90
<b>第5章 結論</b>		<b>92</b>
謝辞		94
参考文献		95
付録 A プログラムブロックで作成可能なシナリオ		99
付録 B マーカレストラッキング		106
付録 C シナリオファイルのフォーマット		108
付録 D 本評価で用いた資料		110

## 目 次

2.1	AR で CG を重畳表示した例 <sup>[11]</sup> . . . . .	3
2.2	消火を体験できる AR 訓練の例 <sup>[12]</sup> . . . . .	4
2.3	津波を体験できる AR 訓練の例 <sup>[13]</sup> . . . . .	4
3.1	MESH <sup>[20]</sup> の電子タグと使用イメージ . . . . .	9
3.2	ビジュアルプログラミングの例 <sup>[18]</sup> . . . . .	9
3.3	ARTM の概要図 . . . . .	10
3.4	漏水の CG で漏水の発生を模擬するイメージ . . . . .	12
3.5	防護服の視界を CG で模擬するイメージ . . . . .	12
3.6	制限時間を表示するイメージ . . . . .	13
3.7	火および煙の CG で火災の発生を模擬するイメージ . . . . .	14
3.8	火および煙の CG で火災が広がる様子を模擬するイメージ . . . . .	14
3.9	爆発の発生を爆発の CG で模擬するイメージ . . . . .	15
3.10	消火を模擬するイメージ . . . . .	15
3.11	侵入者がフェンスを乗り越える様子を CG で模擬するイメージ . . . . .	16
3.12	侵入者が爆弾を設置する様子を CG で模擬するイメージ . . . . .	17
3.13	警察官が侵入者を逮捕する様子を CG で模擬するイメージ . . . . .	17
3.14	他の訓練体験者の CG を重畳表示するイメージ . . . . .	18
3.15	条件と動作でシナリオの一部を作成する様子 . . . . .	24
3.16	ベースブロックとパーツブロックで条件ブロックを作成する様子 . . . . .	25
3.17	ベースブロックとパーツブロックで動作ブロックを作成する様子 . . . . .	26
3.18	シナリオブロックを作成する様子 . . . . .	26
3.19	複数の動作ブロックを組み合わせる様子 . . . . .	27
3.20	「特定変数」のパーツブロックの種類 . . . . .	28
3.21	数字のパーツブロックの種類 . . . . .	28
3.22	条件ベースブロックの種類 . . . . .	28
3.23	オブジェクトパーツブロックの種類 . . . . .	29
3.24	「動き」のパーツブロックの種類 . . . . .	29

3.25	動作ベースブロックの種類	29
3.26	パーツブロックをベースブロックの間違った箇所にはめ込めない様子	30
3.27	プログラムブロック一覧	30
3.28	訓練環境構築サブシステムの使用時の流れ	31
3.29	ヘッドマウント型ディスプレイの例 <sup>[23]</sup>	34
3.30	プロジェクション型ディスプレイの例 <sup>[24]</sup>	34
3.31	ハンドヘルド型ディスプレイの例 <sup>[25]</sup>	35
3.32	訓練環境構築サブシステムのシステム構成	36
3.33	クライアントアプリの処理の流れ	37
3.34	構築サーバアプリの処理の流れ	38
3.35	トラッキング失敗を示した合成画像を表示するイメージ	39
3.36	認識したシナリオブロック上に緑色のハイライトを表示するイメージ	39
3.37	訓練環境体験サブシステムの使用時のイメージ	40
3.38	訓練環境体験サブシステムのシステム構成	42
3.39	体験サーバアプリの処理の流れ	43
3.40	訓練環境構築サブシステムのハードウェア構成	44
3.41	本システムで用いた高性能ノート PC の外観	45
3.42	本システムで用いたタブレット PC の外観	45
3.43	本システムで用いた RGB-D カメラの外観	46
3.44	本システムで使用した ArUco マーカ <sup>[33][34]</sup> の外観	48
3.45	ArUco マーカおよびそれを貼り付けた各パーツブロックの大きさ	49
3.46	ArUco マーカを貼り付けたベースブロックの大きさ	49
3.47	シナリオブロックの外観のイメージ	51
3.48	複数のシナリオブロックが画像上にある様子	52
3.49	各マーカの距離によってグルーピングする様子	53
3.50	上部 2 つのみグルーピングする様子	53
3.51	動作ブロックが配置される範囲を切り出す様子	54
3.52	PnP 問題を解くイメージ	55
3.53	シナリオブロックの姿勢を計測するイメージ	55
3.54	数字パーツブロックが配置される領域を切り出す様子	57
3.55	3D プリンタで印刷したパーツブロックの外観とその大きさ	58
3.56	3D プリンタで印刷したベースブロックの外観とその大きさ	58

3.57	ArUco マーカを貼り付けたベースブロックの例 . . . . .	59
3.58	ArUco マーカを貼り付けたパーツブロックの例 . . . . .	59
3.59	訓練環境体験サブシステムのハードウェア構成 . . . . .	60
3.60	本システムで用いた Bluetooth スピーカの外観 . . . . .	61
3.61	訓練環境体験サブシステムで実行するプログラムとデータの流れ . . . . .	62
3.62	実空間のカメラと同時に仮想カメラが移動し合成画像を作成する様子 . . . . .	64
3.63	オクルージョンを考慮していない画像としている画像のイメージ . . . . .	64
3.64	Zバッファを用いて合成画像を作成する様子 . . . . .	65
3.65	レンダリングアプリの処理の流れ . . . . .	66
3.66	サウンドアプリの処理の流れ . . . . .	67
3.67	シナリオブロック撮影時の画面の例 . . . . .	68
3.68	火の CG を重畳表示した画面 . . . . .	68
3.69	移動する人の CG を重畳表示した画面 . . . . .	69
4.1	評価に使用した部屋の様子 . . . . .	74
4.2	評価に使用した部屋の見取り図 . . . . .	74
4.3	本評価の手順 1 . . . . .	76
4.4	本評価の手順 2 . . . . .	77
4.5	評価者がプログラムブロックを組み立てる様子 . . . . .	80
4.6	評価者がシナリオブロックを設置する様子 . . . . .	80
4.7	評価者がシナリオブロックを撮影する様子 . . . . .	80
4.8	評価に使用したプログラムブロック . . . . .	81
4.9	訓練シナリオのタイムラインのイメージ . . . . .	86
4.10	距離条件の範囲をハイライト表示するイメージ . . . . .	87
4.11	プレビュー機能のイメージ . . . . .	88
A.1	プログラムブロックで作成可能なシナリオ 1 . . . . .	99
A.2	プログラムブロックで作成可能なシナリオ 2 . . . . .	100
A.3	プログラムブロックで作成可能なシナリオ 3 . . . . .	101
A.4	プログラムブロックで作成可能なシナリオ 4 . . . . .	102
A.5	プログラムブロックで作成可能なシナリオ 5 . . . . .	103
A.6	プログラムブロックで作成可能なシナリオ 6 . . . . .	104
A.7	プログラムブロックで作成可能なシナリオ 7 . . . . .	105

B.1	特徴点ベースのマーカレストラッキングの処理の流れ	106
B.2	2枚の画像における特徴点検出の結果の例	107
B.3	2枚の画像における特徴点マッチングの結果の例	107
C.1	3Dモデル配置のフォーマットと例	108
C.2	時間条件のフォーマットと例	108
C.3	時間条件で効果音を再生する際のフォーマットと例	108
C.4	人物が移動する際のフォーマットと例	109
C.5	距離条件のフォーマットと例	109
C.6	距離条件で効果音を再生する際のフォーマットと例	109
D.1	シナリオ作成課題1の課題内容を記した資料	110
D.2	シナリオ作成課題2の課題内容を記した資料	111
D.3	自由作成課題の課題内容を記した資料	112
D.4	評価概要の事前説明に用いた資料	113
D.5	ARTMの説明ビデオのキャプチャ画像	114
D.6	説明ビデオ内容の補足資料	115
D.7	プログラムブロックの説明資料1	116
D.8	プログラムブロックの説明資料2	117
D.9	プログラムブロックの説明資料3	118
D.10	プログラムブロックの説明資料4	119
D.11	プログラムブロックの説明資料5	120
D.12	ARTMの容易さに関するアンケート用紙	121
D.13	訓練シナリオの自由度に関するアンケート用紙	122



# 表 目 次

3.1	プログラムブロックで実現する訓練要素 . . . . .	19
3.2	プログラムブロックに採用したオブジェクトの一覧 . . . . .	20
3.3	Scratch の基本的な機能群一覧 . . . . .	21
3.4	プログラムブロックに採用した機能と採用しなかった機能 . . . . .	23
3.5	本システムで用いた高性能 PC の仕様 . . . . .	45
3.6	本システムで用いたタブレット PC の仕様 . . . . .	46
3.7	本システムで用いた RGB-D カメラの仕様 . . . . .	46
3.8	本システムで用いたルータの仕様 . . . . .	47
3.9	マーカの分類 . . . . .	50
3.10	マーカの ID とプログラムブロックの種類に対応 . . . . .	51
3.11	MakerBot Replicator + の仕様 . . . . .	57
3.12	本システムで用いた Raspberry Pi の仕様 . . . . .	61
3.13	本システムで用いた Bluetooth スピーカの仕様 . . . . .	61
4.1	容易さに関するアンケート項目と ARTM の容易さに関する項目の対応	78
4.2	訓練シナリオの自由度に関するアンケート項目と訓練シナリオの自由度 に関する項目の対応 . . . . .	78
4.3	評価者の属性 . . . . .	79
4.4	ARTM の容易さに関するアンケートの結果 . . . . .	81
4.5	ARTM の容易さに関するアンケートの結果および回答理由 (X)-1 から (X)-10 . . . . .	82
4.6	容易さに関するアンケートの結果および回答理由 (X)-11 から (X)-13 . .	83
4.7	自由作成課題で評価者が考えた AR 訓練 . . . . .	83
4.8	訓練シナリオの自由度に関するアンケートの結果 . . . . .	83
4.9	訓練シナリオの自由度に関するアンケートの結果および回答理由 . . . .	84
4.10	評価者 g の自由記述欄への回答 . . . . .	84

# 第 1 章 序論

2011 年の東北地方太平洋沖地震における福島第一原子力発電所での事故を受け、日本の発電所および原子力関連施設などの重要施設における安全性の見直しが進んでいる。例えば、2018 年に原子力災害対策特別措置法が改正され、災害対策が大幅に強化されている<sup>[1]</sup>。これによって津波の衝撃を緩和する防波堤のかさ上げや<sup>[2]</sup>、外部火災の影響を防ぐ防火帯の確保<sup>[3]</sup>をはじめとする、様々な緊急事態を想定した対策が実施されている。こうした災害の直接的な外力を構造物によって物理的に軽減するハード面の対策が進んでいる一方、情報通信の強化や防災教育の充実などのソフト面の対策も進んでいる。

重要施設におけるソフト面の対策の 1 つとして、防災訓練が挙げられる。実際に、全国の発電所や原子力関連施設などの重要施設では、巨大地震などの非常災害を想定した防災訓練が実施されている<sup>[4][5]</sup>。また、原子力関連施設では防災訓練だけでなく、警察と協力し、施設に対するテロを想定した不法事案対策訓練も実施されている<sup>[6]</sup>。

こういった防災訓練および不法事案対策訓練の方法は、机上訓練、シミュレーション訓練および実働訓練の 3 種類に分けられる<sup>[7]</sup>。机上訓練は、机上で緊急時のマニュアルに従って作業を進める訓練であり、必要最小限の人数で実施可能である。シミュレーション訓練は災害時に発生する様々な状況を想定し、実際に対応できるかを訓練で確認する訓練である。最後の実働訓練は、訓練項目のすべてを実際に行動することによって確認する訓練である。重要施設の現場で実施されている訓練は、シミュレーション訓練か実働訓練のどちらかに分類されることが多い。

こうしたシミュレーション訓練および実働訓練は、訓練参加者が実際に行動するため、机上訓練だけでは確認できなかった問題を発見できるという利点がある一方で、実施できる日数が限られてしまうという問題点がある。実際に、原子力発電所では年に 1、2 度しか訓練が実施されていない<sup>[5]</sup>。また、訓練当日も通常通りに施設の運営を続ける場合には、関係者全員が訓練に参加できるわけではない。そのため本来は参加すべき人が訓練に参加できない可能性がある。さらに、多くの人と同時に参加する訓練に費やすことができる時間は限られているため、体験できる訓練シナリオも 1 つ程度に限られてしまうという問題がある。

このように現場で実施される訓練には、参加できる人および体験できる訓練シナリ

オが限られるという問題点ある。ソフト対策のさらなる強化のためには、誰もが空き時間に手軽かつ効果的に様々な訓練を受けられる環境の実現が必要である。

一方、近年 Augmented Reality を用いた訓練（以下、AR 訓練）が注目されている<sup>[8][9]</sup>。AR とは、視覚、聴覚および触覚などの知覚に与えられる情報を、コンピュータによる処理で実世界に追加する技術である<sup>[10]</sup>。AR 訓練とは、従来の訓練においては実現することが難しい火災や津波などの現象を AR を用いて実現した訓練である。AR 訓練においては、ユーザはまるで本当に火災や津波が発生しているように感じられる。こういった AR 訓練は、必要な機器さえ揃えてしまえば、現場で簡便に 1 人で実施できることが多い。そのため、重要施設の防災訓練および不法事案対策訓練にも AR 訓練を取り入れることがソフト対策向上に効果的であると考えられる。しかし AR 訓練環境の構築には様々な専門知識が必要であるため、専門外である重要施設の作業員が AR 訓練環境を構築することは難しいという問題がある。

そこで、本研究では、AR 訓練環境の構築に必要な専門知識を持っていない重要施設の作業員でも容易に AR 訓練環境を構築できるシステムを開発することを目的とする。

本論文は、本章を含めた全 5 章で構成されている。第 2 章では、研究の背景として、AR 訓練の現状と課題を述べ、本研究の目的を述べる。第 3 章では、本研究で開発した AR 訓練環境構築システムについて述べる。第 4 章では本システムの評価の方法および結果を述べ、結果に対する考察を述べる。最後に、第 5 章では本研究の結果をまとめ、今後の展望を述べる。

## 第 2 章 研究の背景と目的

本章ではまず、AR 訓練の現状と問題点について述べる。次いで、AR 訓練環境の構築に関する既往研究について述べる。最後に本研究の目的と意義について述べる。

### 2.1 AR 訓練の現状と問題点

AR では、図 2.1 に示すように、ユーザの視界に、文字や図形などの Computer Graphics（以下、CG）を重畳表示することで、ユーザが得る実世界空間の情報を拡張する。近年この AR を用いた訓練が注目されている [8][9]。

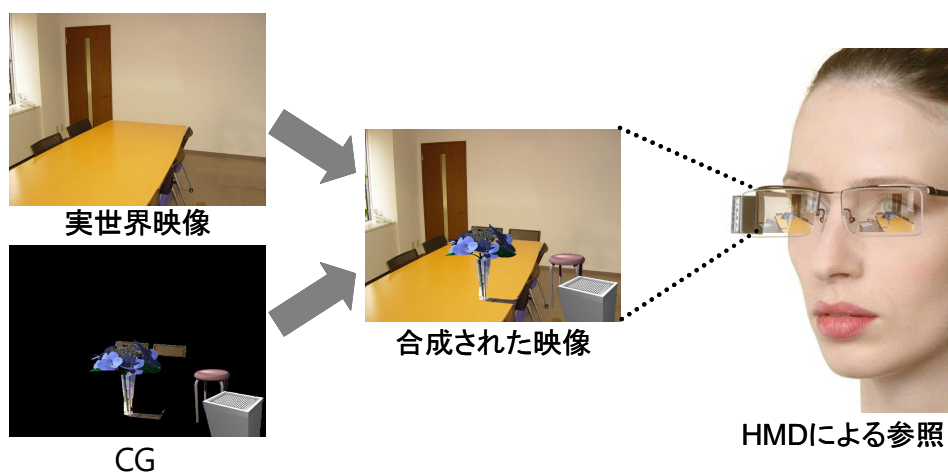


図 2.1: AR で CG を重畳表示した例 [11]

板宮ら [12] は部屋内で起こった火災の消火作業を体験できる AR 消火訓練アプリを開発した。図 2.2 に AR 消火訓練アプリの画面例を示す。このように AR 訓練では、従来の訓練で実現が難しい火災などのイベントを実現できる。また、実働訓練と同じく、現場でユーザが実際に行動する訓練を実施できる。

また、入江ら [13] は図 2.3 に示すように、スマートフォン型ヘッドマウントディスプレイに AR で津波などの災害状況を提示し、訓練参加者に避難行動を選択させる避難訓練システムを試作した。本システムでは津波の CG を用いて臨場感のある災害状況を提示し、ユーザの緊迫感を高めると同時に、ユーザの選択によりシナリオの進行が変化するインタラクティブな訓練の実現を目指している。

これらのAR訓練は、多くの事前準備を必要としない、ユーザ1人で体験できる、従来の訓練では実現することが難しいイベントを実現できる、従来の訓練と同様に現場で体験できる、という利点がある。そのため重要施設における訓練に取り入れることで、重要施設における安全性のソフト対策として効果的だと考えられる。



図 2.2: 消火を体験できる AR 訓練の例<sup>[12]</sup>



図 2.3: 津波を体験できる AR 訓練の例<sup>[13]</sup>

しかし一方で、こういったAR訓練環境の構築にはプログラミングや特殊なソフトウェアに関する専門知識が必要である。そのため、AR訓練環境を構築する際には、ソフトウェア開発などを専門とする企業に外注することが多い。しかし、ソフトウェア開発を専門としている企業は、現場の状況や訓練内容に精通しているわけではないため、構築されたAR訓練環境が、現場で求められるAR訓練環境と乖離してしまう可

能性がある。また、外注の場合は、作成できる訓練シナリオの種類が限られてしまうため、訓練シナリオのバリエーションが少なくなってしまうことも問題である。こういった問題が生じないように、現場の状況と訓練内容に最も精通した重要施設の作業員が、現場で様々な状況を想定して自ら AR 訓練環境を構築できるようにすることが望まれる。

また、PC 上で AR 訓練環境を構築する場合、現場の状況を参照することが困難である。そのため AR 訓練環境の構築と訓練シナリオの考案を同時並行で行うことは難しい。例えば、火災の発生場所を変更することにより、避難経路を変化させる場合、現場で訓練環境を見ながら行うほうが容易だと考えられる。現場の図面を見ながら火災の発生場所を設定することも考えられるが、図面では正確な 3 次元位置を設定することは難しい。このような問題が生じないように、現場で訓練環境を直接観察しながら、訓練環境を構築することが望まれる。

上記のように、現状、AR 訓練環境の構築にあたっては、専門知識が必要であること、現場を直接観察しながらの作業が困難であることが課題となっている。

## 2.2 AR 訓練環境構築に関する既往研究

本節では AR 訓練環境を構築するシステムに関する既往研究について述べる。AR 訓練に関する先行研究や実用例は多く存在する<sup>[14][15]</sup>が、AR 訓練環境を構築するシステムに関する研究は少ない。

板宮らは、3DOF(Degree of Freedom) コントローラの操作でバーチャル発火点を現実空間の任意の場所に配置できる AR 消火訓練アプリを開発した<sup>[12]</sup>。板宮らはこれによって容易に実環境において AR 火災訓練を実施できると述べている。

板宮らは、他にも、津波や洪水発生時の浸水状況および火災煙状況を体験できる AR 災害疑似体験アプリ、Disaster Scope を開発した<sup>[16]</sup>。このアプリでは、浸水状況を AR で模擬することができるが、その水位を 10cm 毎に設定できる。また、水位を段々と連続的に上昇させることができ、その上昇速度を任意に設定できる。

これらのアプリでは GUI (グラフィカルユーザインタフェース) 上でコントローラや、数値を用いて、現場を直接観察しながら AR 訓練環境を構築できるが、複雑な訓練シナリオを構築することは難しいという問題がある。

## 2.3 研究の目的と意義

2.1節で述べたように、AR 訓練環境の構築には、主にプログラミングに関する専門知識が必要であるという課題と、現場での構築が困難であるという課題がある。本研究では、それらの課題に注目し、実世界指向プログラミングを用いた AR 訓練環境構築システムを開発することを目的とする。本研究では主にプログラミングに関する専門知識を持っていない人でも容易に理解および操作でき、かつ現場で扱えるように、GUI を用いない実世界指向プログラミング<sup>[17]</sup>を用いた。既往研究のように GUI 上で AR 訓練を構築するシステムは存在するが、実世界指向プログラミングを用いて AR 訓練環境を構築することを可能にした研究は見当たらない。実世界指向プログラミングの詳細については3.2節で述べる。本研究により、AR 訓練構築に適したインタフェース設計に知見を加え、今後 AR 訓練環境構築システムを開発する際のインタフェース設計の指針の1つになると期待する。また、それによって AR 訓練環境構築の設計が容易になり、重要施設における安全性のソフト面対策に寄与できると期待できる。

## 第 3 章 AR 訓練環境構築システム（ARTM） の開発

本章では、第 2 章で述べた AR 訓練作成における課題の解決を目指して開発した、AR 訓練環境構築システム（Augmented Reality Training Maker; ARTM）について述べる。はじめに、システムの概略設計について述べ、次に、本システム実現のために新たに考案したプログラミング言語である「プログラムブロック」の設計について述べる。その後、本システムを構成する 2 つのサブシステムの設計について述べ、最後に実装したシステムについて述べる。

### 3.1 システムの目標

第 2 章で述べたように、本来であれば AR 訓練環境は訓練内容に精通した現場の作業員によって作成されることが望ましいため、プログラミングなどの高度な情報系のスキルを持っていない現場作業員でも容易に構築可能にする必要がある。また、現場を直接観察しながら AR 訓練環境を構築できることが望まれる。さらに、作成できる訓練シナリオは、現場で求められている AR 訓練を実現するのに十分な自由度で作成できる必要がある。

以上の理由から、本研究で開発するシステムの目標を以下の 3 つに定める。

- プログラミングの高度なスキルを持っていない人でも容易に理解および操作ができること
- 訓練を実施する現場を直接観察しながら訓練環境を構築できること
- 現場で求められている水準の複雑さの訓練シナリオを作成できること

### 3.2 ARTM の概略設計

本研究では 3.1 節で述べた目標を達成するために、プログラミングの高度なスキルを持っていない人でも容易に使用できる新しいプログラミング言語を開発した。



新しいプログラミング言語を開発するにあたって、実世界指向プログラミング<sup>[17]</sup>とビジュアルプログラミング<sup>[18]</sup>を参考にした。以下ではこの2つのプログラミング方法とこれらを参考にした理由について述べる。

### (1) 実世界指向プログラミング

多くのプログラミング言語は、テキストとして記述することを想定している。また、プログラムで扱う対象として主に文章や画像を想定しており、テキストや数で容易に表現できる対象を扱う場合がほとんどであった<sup>[17]</sup>。このように、プログラムとその扱う対象の表現形式が同じ場合はユーザにとって理解しやすい。しかし両者の表現形式が大きく異なる場合は、扱う対象が持つ情報とテキスト表現とのギャップが大きくなり、ユーザにとって理解が困難になる<sup>[17]</sup>。例えば3次元CGを用いたゲームなどを開発する際に広く使われているゲームエンジンであるUnity<sup>[19]</sup>では、CGの3次元位置および向きを指定する場合、

```
transform.SetPositionAndRotation(10,11,10,1,0,0,0);
```

のように表現する。このようなテキスト表現は、3次元空間内に存在する物の位置や向きなど対象が持つ情報とのギャップが大きい。実世界指向プログラミングは、このようなギャップを最小限に抑えるために、実世界の事物のみを使って行うプログラミング方法である<sup>[17]</sup>。

増井<sup>[17]</sup>は実世界の事物を扱うプログラミングは実世界の事物のみを使って行えるほうがはるかに都合がよいと主張しており、AR訓練のように実世界の事物を扱うコンテンツのプログラミングを行うには、実世界指向プログラミングが適していると考えられる。

実世界指向プログラミングの例として、図3.1に、ロボット教育の教材「MESH」<sup>[20]</sup>を示す。MESHでは、スマートフォンやタブレット上で実行可能なソフトウェア開発キットを用いて、実世界のあらゆる事物に取りつけた電子タグで計測した動きや温度を使ったプログラミングが可能である。例えば、1人暮らしの高齢者の安否を確かめるために、日常的に利用する食器棚の扉に電子タグを張り付けることで、扉の開閉の動きを感知した際にスマートフォンにメールが届くシステムを開発できる。

このように、従来のARコンテンツ作成時のようにテキストやGUI（グラフィカルユーザインタフェース）で実世界の事物を設定することは、表現形式のギャップが存在するため難しいが、MESHのように実世界の事物に電子タグのような目印となるものを設置する方法を用いることで、プログラミング初心者でも容易に実世界の事物を設定できるようになると考えられる。



図 3.1: MESH<sup>[20]</sup> の電子タグと使用イメージ

## (2) ビジュアルプログラミング

ビジュアルプログラミングとは、図 3.2 に示すように、テキストで記述するのではなく、GUI 上のマウス操作でブロックを配置することにより、プログラミングを行う方法である。予め設定された基本的な関数に相当するブロックを組み合わせることで、目的のプログラムを作成できる。ビジュアルプログラミングはプログラミング教育を主目的として作られた方法で、プログラミング初心者でも直感的に理解できるように設計されている。

ビジュアルプログラミングのように予め設定された関数のブロックを組み合わせる方法を用いれば、プログラミング初心者でも容易にプログラミングを行えると考えられる。

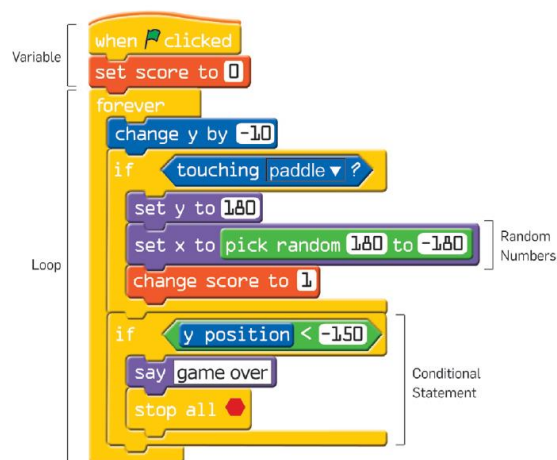


図 3.2: ビジュアルプログラミングの例<sup>[18]</sup>

上記2つのプログラミング方法を参考に、実世界でブロックを組み合わせることでよりプログラミングを行うプログラミング言語「プログラムブロック」を開発する。

本研究では、このプログラムブロックの言語仕様を定めるとともに、AR 訓練環境のシナリオ作成と訓練体験が可能な AR 訓練環境構築システム ARTM を開発する。

図 3.3 に、ARTM の概要図を示す。ARTM は、プログラムブロックを用いて訓練シナリオを作成する訓練環境構築サブシステムと、訓練環境構築サブシステムで作成した訓練シナリオを体験する訓練環境体験サブシステム、の2つのサブシステムで構成する。訓練環境構築サブシステムでは、訓練環境構築者はプログラムブロックを組み合わせ環境中に設置し、それらを認識デバイスを用いて認識させることで、訓練シナリオを作成できる。訓練環境体験サブシステムでは、訓練体験者は表示デバイスを通して訓練シナリオを体験できる。

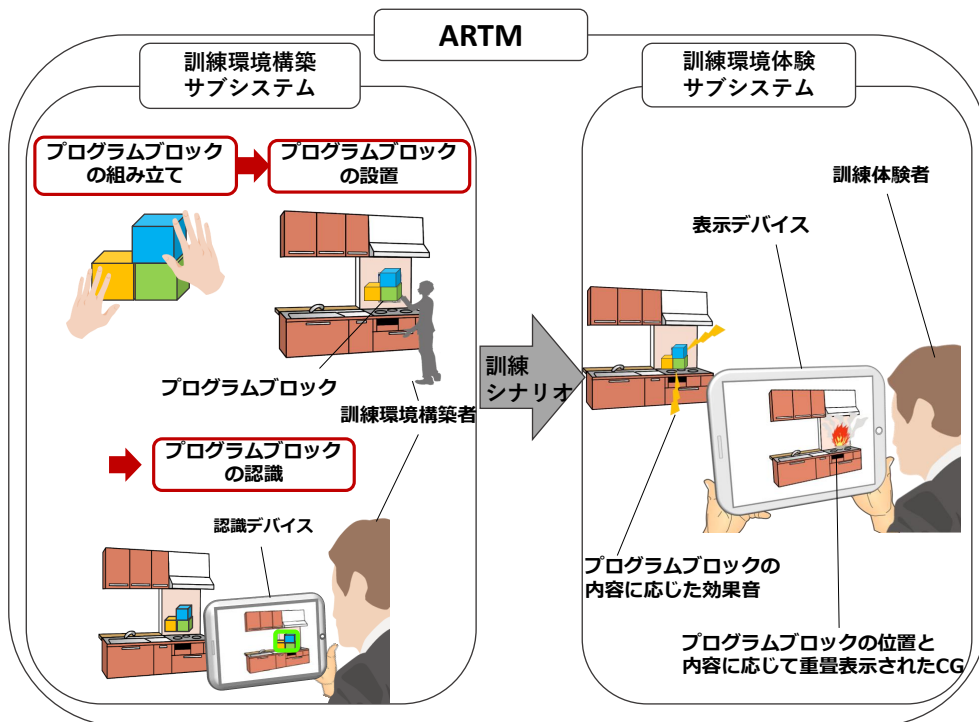


図 3.3: ARTM の概要図

## 3.3 プログラムブロックの設計

### 3.3.1 プログラムブロックで実現する訓練要素の検討

プログラムブロックは、従来のビジュアルプログラミングのように、関数に相当するブロックを組み合わせてプログラミングを行うプログラミング言語とする。その際、ブロックの種類が少なければ、全体を把握することが容易になるが、プログラムブロックによって実現できる訓練シナリオの自由度が低くなってしまふ。しかし、ブロックの種類が多ければ、訓練シナリオの自由度は高くなる一方で、組み合わせの複雑さが増し、ユーザの理解および操作が難しくなってしまう。そのため、容易さと自由度を両立するためには、現場で求められている訓練シナリオを実現するのに適した種類のプログラムブロックに限って準備する方が望ましい。本研究では、プログラムブロックの種類を選定するにあたり、実際に重要施設で実施されている訓練を参考にする。

重要施設で実施されている主な訓練として、災害を想定した防災訓練と、テロ行為を想定した不法事案対策訓練が挙げられる。この2種類の訓練を参考に、AR訓練の要素を挙げ、その中からプログラムブロックで実現する訓練要素を検討した。以下に、その詳細を述べる。

#### (1) 防災訓練

重要施設で実施されている主な防災訓練として、地震訓練が挙げられる。ある原子力関連施設では、地震により原子炉が損傷し、冷却水が漏れ出す状況を想定した訓練が実施されている。作業員は防護服と酸素ボンベを装着し、漏水の発生箇所を探し当て、放射能汚染の観点で安全に作業できる時間内に安全な場所に戻ってくるという訓練である。

この訓練をARで実現する場合、図3.4に示すように、訓練開始時から漏水のCGおよび効果音を再生し続けることで漏水の発生を模擬できる。また、図3.5に示すように、防護服をかぶった際の視界をARで模擬することも考えられる。他に、図3.6に示すように、作業可能な残り時間を表す制限時間を表示することも考えられる。

図3.4の漏水の発生は、従来の訓練では施設に損傷を与える可能性があるため模擬することが困難であるが、AR訓練ではCGを用いて模擬でき、訓練の臨場感を高める観点で効果的であると考えられるため、プログラムブロックで実現する訓練要素として採用した。図3.5で示した災害時に視界が狭まるという状況は、ARを用いて実現するよりも、実際に防護マスクを被る方が、容易かつより高い臨場感で模擬できるため、プログラムブロックで実現する訓練要素として採用しなかった。図3.6で示した作業の

制限時間に関しては、実際の緊急時には参照できない情報であり、実現することで逆に訓練の現実感を損ねてしまうと考えられるため、今回はプログラムブロックで実現する訓練要素として採用しなかった。

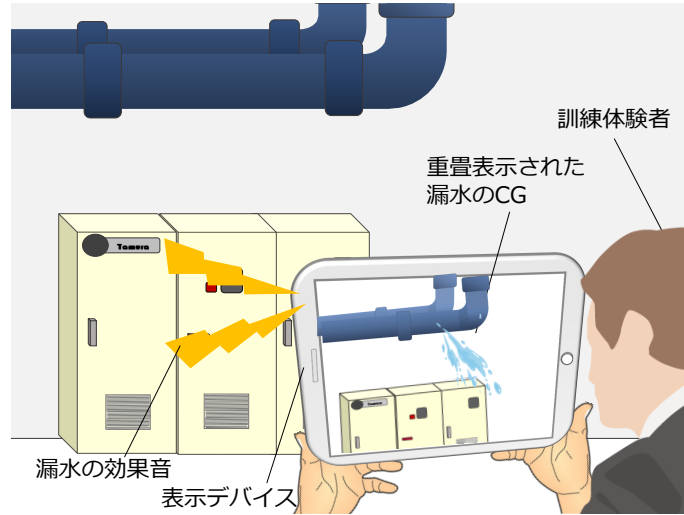


図 3.4: 漏水の CG で漏水の発生を模擬するイメージ

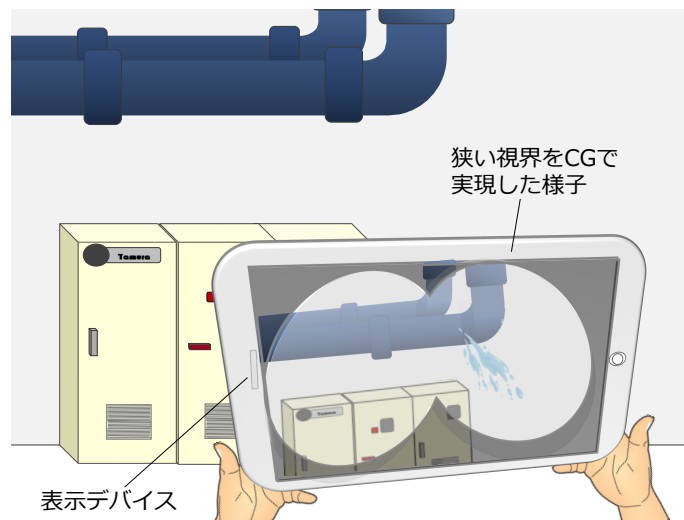


図 3.5: 防護服の視界を CG で模擬するイメージ

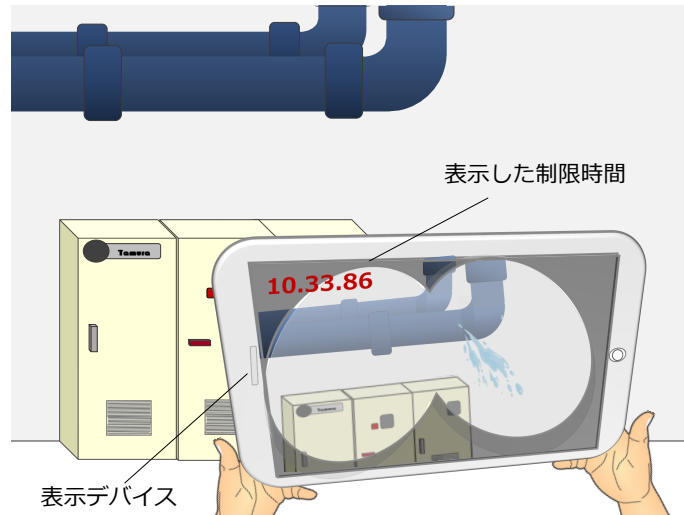


図 3.6: 制限時間を表示するイメージ

重要施設では、新潟県中越沖地震による原子力発電所の火災を受け、火災訓練も重要視されており、地震による衝撃でタービンなどの機器から発火することを想定した訓練などが実施されている。これらの訓練では、作業員は消火器を用いた消火活動や、適切な避難経路で避難することが求められる。従来の訓練では発火、火が燃え広がる様子およびそれによって爆発が起こる様子は実現が難しいため、訓練体験者はそれらが起こったと頭の中だけで想定して、訓練を模擬することが多い。

この訓練を AR で実現する場合、図 3.7 に示すように、訓練開始後の適切なタイミングで火や煙の CG および効果音を再生することで、火災の発生を模擬できる。また、図 3.8 に示すように、時間経過により火や煙の CG を複数表示することや、大きさを変化させることによって、火が燃え広がる様子を模擬できる。火が燃え広がった際の爆発の発生も、図 3.9 に示すように、爆発の CG および効果音を再生することで模擬できる。消火器を用いて消火する様子は、図 3.10 に示すように、消火器および消火剤の CG を重畳表示し、消火活動が行われた際には火の CG を非表示にすることで模擬できる。

火災の発生、爆発の発生および時間経過によって火が燃え広がる様子は、漏水と同様に従来の訓練方法では模擬が困難であるが、AR 訓練では CG を用いて模擬でき、臨場感を高めると考えられるため、プログラムブロックで実現する訓練要素に採用した。一方消火の様子を消火器と消火剤の CG で実現する場合には、CG 同士の接触判定など複雑な設定が必要であるため、プログラムブロックで実現することは難しい。そのため、ユーザが近づくことで火が鎮火する様子を、プログラムブロックで実現できるようにする。

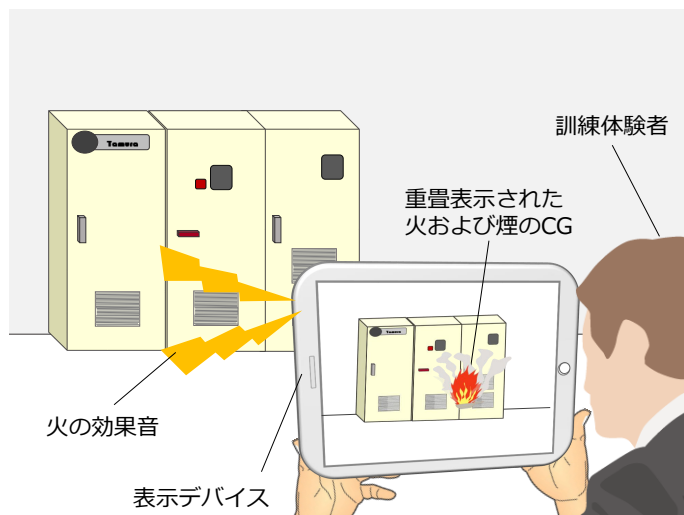


図 3.7: 火および煙の CG で火災の発生を模擬するイメージ

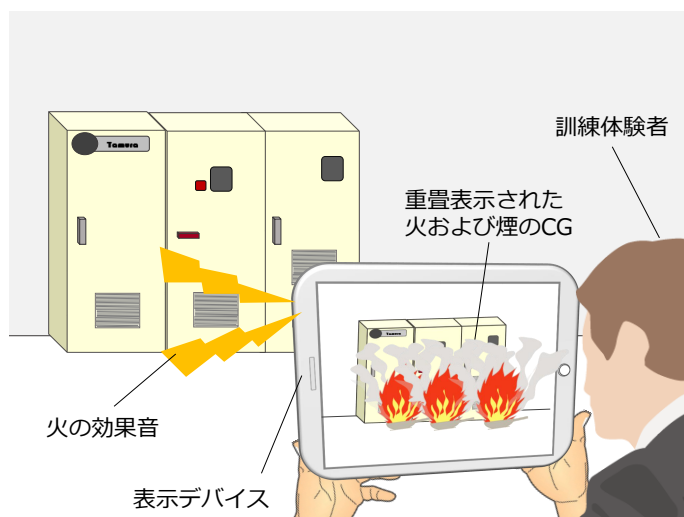


図 3.8: 火および煙の CG で火災が広がる様子を模擬するイメージ

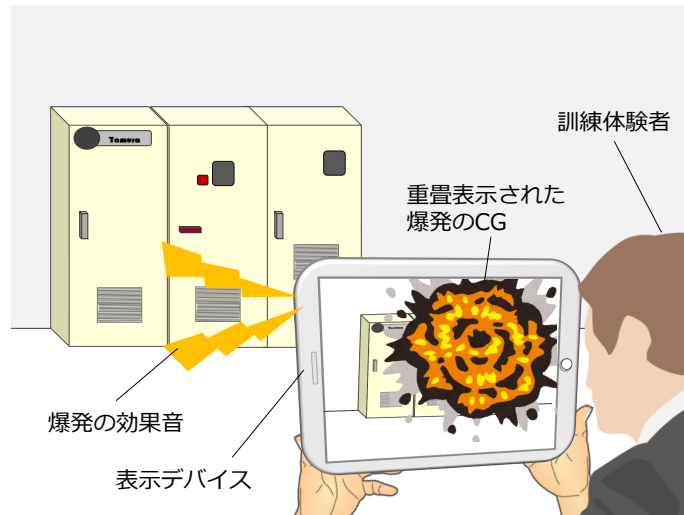


図 3.9: 爆発の発生を爆発の CG で模擬するイメージ

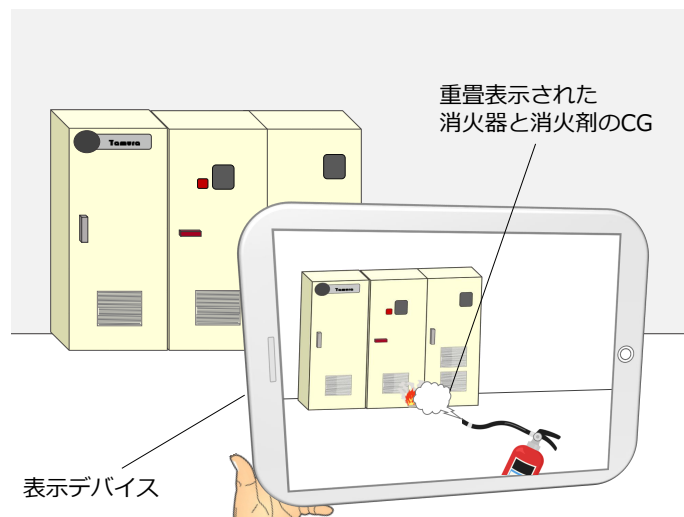


図 3.10: 消火を模擬するイメージ

## (2) 不法事案対策訓練

重要施設で実施されている不法事案対策訓練としては、侵入者対策訓練が挙げられる。ある原子力関連施設では、テロ攻撃行為だけでなく、侵入者による放射性物質の持ち出しを想定した訓練も実施されている。

訓練シナリオの例としては以下のようなものである。放射性物質の持ち出しを目論む侵入者がフェンスを乗り越え、施設内に侵入する。侵入者は施設内の防護扉に爆弾を仕掛け、爆破し、防護扉内に侵入する。それらを目撃した訓練体験者は警察に通報



し、状況を伝える。その後施設に到着した警察官が侵入者を逮捕する。訓練体験者は警察に侵入者の位置や特徴を正確に伝え、適した避難経路で避難することが望まれる。

このような訓練を実施する際には、現場の作業員が侵入者役を演じるなど、本来の役割と異なる役割を演じる場合などもあるが、実際に警察に協力を要請して本物の警察官が訓練に参加する場合などもある。

このような訓練を AR 訓練で模擬する場合、従来の訓練では現場の作業員が演じていた侵入者を CG で模擬できる。また、図 3.11、図 3.12 に示すように、侵入者がフェンスを乗り越える様子や爆弾を設置する様子を、侵入者の CG で模擬できる。図 3.9 と同じく、設置した爆弾が爆発する様子も CG および効果音を用いて模擬できる。作業員が通報し、警察に状況を伝える様子は、電話の音や警察の受け答えの音声を流すことで模擬できる。また、従来の訓練においては協力を要請していた警察官も、CG で模擬できる。さらに、通報後に警察官が到着する様子を、警察官の CG で模擬できる。図 3.13 に示すように、最後に警察が侵入者を捕まえる様子も CG で模擬できる。



図 3.11: 侵入者がフェンスを乗り越える様子を CG で模擬するイメージ



図 3.12: 侵入者が爆弾を設置する様子を CG で模擬するイメージ



図 3.13: 警察官が侵入者を逮捕する様子を CG で模擬するイメージ

侵入者などを CG で模擬することで、従来の訓練のように現場の作業員や警察官の協力なしで訓練を実施できるようになり、訓練の実施が容易になるという利点が生じるため、プログラムブロックで実現する訓練要素に採用した。

また、訓練体験者の行動は、訓練を同時に体験している他者に影響されるため<sup>[21]</sup>、訓練は複数人で体験することが望ましい。複数人同時に参加することが難しい状況でも AR 訓練を体験できるようにするためには、他の訓練体験者も AR で実現することが考えられる。図 3.14 に示すように、他の訓練体験者の CG を重畳表示することで、訓練を同時に体験する他者が動いている様子を模擬できる。

このように他の訓練体験者を CG で実現することで、ユーザの行動に影響を与えるシナリオを作成できるようになり、訓練シナリオの自由度が高くなると考えられるため、プログラムブロックで実現する訓練要素に採用した。他の訓練体験者は、防災および不法事案対策訓練の両方に用いられるため、普遍的な訓練要素であり様々な種類の訓練で使用できると考えられる。その際、他の訓練体験者が避難する様子を実現可能にするために、時間経過によって他の訓練体験者が移動する様子も訓練要素に採用した。

時間経過によって侵入者が移動したりフェンスを乗り越えたりする様子や移動する様子は、訓練シナリオの自由度を向上させると考えられるため、プログラムブロックで実現する訓練要素に採用した。爆弾を侵入者が持っている様子を表現することは、訓練における主要素ではないため、プログラムブロックで実現する訓練要素に採用しなかった。何かを設置するという動作は、訓練シナリオの自由度を向上させると考えられるため、侵入者がかがむ様子をプログラムブロックで実現する訓練要素に採用した。訓練体験者が通報作業を実施可能にすることは、訓練におけるユーザのインタラクティブ性を高め、訓練に参加しているという感覚を効果的に誘発できると考え、採用した。

警察官を CG で実現することで、従来の訓練のように警察官の協力なしで訓練を実施できるようになり、導入効果が高いため、プログラムブロックで実現する訓練要素に採用した。警察が侵入者を逮捕する様子は、警察官と侵入者の設定位置に応じたりリアルな CG を合成することが技術的に難しい上に、訓練における主要素ではないため、プログラムブロックで実現する訓練要素に採用しなかった。

以上の検討を元に決定したプログラムブロックで実現する訓練要素を、表 3.1 に示す。

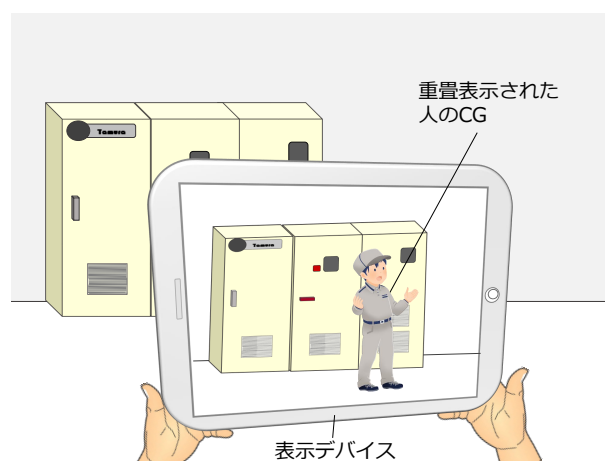


図 3.14: 他の訓練体験者の CG を重畳表示するイメージ

表 3.1: プログラムブロックで実現する訓練要素

訓練の種類	プログラムブロックで実現する訓練要素
(1) 防災訓練	(1-a) 訓練開始時における漏水の発生 (1-b) 任意のタイミングにおける火の発生 (1-c) 任意のタイミングにおける煙の発生 (1-d) 時間経過により火が燃え広がる様子 (1-e) 時間経過による爆発の発生 (1-f) ユーザが近づくことにより火が鎮火する様子 (1-g) 他の訓練体験者 (1-h) 時間経過によって他の訓練体験者が移動する様子
(2) 不法事案対策訓練	(2-a) 侵入者 (2-b) 時間経過によって侵入者がフェンスを乗り越える様子 (2-c) 時間経過によって侵入者が移動する様子 (2-d) 侵入者がかがむ様子 (2-e) 爆発の発生 (2-f) ユーザが通報する様子 (2-g) 警察官 (2-h) 通報を受けて警察官が移動する様子

### 3.3.2 プログラムブロックの機能の検討

3.3.1 項で述べた訓練要素を実現するにあたり、プログラムブロックが最低限備えるべき機能を選定する必要がある。そこで、一般的なビジュアルプログラミングに備わっている機能を調べ、その中からプログラムブロックに採用する機能を検討した。以下、その詳細を述べる。

一般的なビジュアルプログラミングとして、教育目的など幅広く用いられている Scratch<sup>[22]</sup> を参考にした。Scratch では、何らかのアクションを起こすオブジェクトが存在し、各オブジェクトは付属したプログラムに従って動く。プログラムは、様々な機能に相当するブロックを組み合わせることによって作成できる。このオブジェクトと、それらを動かすプログラムを作成するためのブロックに相当する機能を選定する必要がある。

はじめに、表 3.1 の訓練要素を実現するのに必要なオブジェクトを検討した。表 3.2

にプログラムブロックに採用したオブジェクトの一覧を示す。

表 3.2: プログラムブロックに採用したオブジェクトの一覧

オブジェクト	
	漏水および漏水の効果音
	火および火の効果音
	煙
	爆発および爆発の効果音
	通報音声
	他の訓練体験者
	侵入者
	警察官

次に、採用したオブジェクトの動きを決定する機能を、Scratch を参考に検討した。Scratch の基本的な機能群を表 3.3 に示す。

例えば、「動き」はオブジェクトの動きを指定する機能群である。「動き」にはオブジェクトを回転させる機能、オブジェクトの2次元座標を変更する機能などがある。「見た目」はオブジェクトの外見を指定する機能群である。「見た目」には色や大きさを変更する機能、オブジェクトの表示非表示を切り替える機能などがある。「音」は音に関する機能群である。「音」には音声を再生および停止する機能や、音量を変更する機能などがある。「特定変数」は予め設定された特定の変数を扱う機能群である。「特定変数」には時間、タイマー、オブジェクトの2次元座標などの特定の数値を取得する変数を扱う機能があり、取得した数値は後述の「制御」の条件として用いられることが多い。「データ」は任意の変数や配列を扱う機能群である。「データ」には新しい変数や配列を作成する機能、配列に変数を追加する機能、配列の長さを取得する機能がある。「演算子」には四則演算する機能や数字の大小を調べる機能があり、「特定変数」と同様に「制御」の条件として用いられる。「特定条件」は予め設定された条件を用いて分岐を設定する機能群である。「特定条件」にはプログラムの開始やオブジェクトとマウスカーソルとの接触などの条件を用いて分岐を設定する機能がある。「制御」はプログラムの流れを制御する機能群である。「制御」には一般的なプログラミング言語における if 文のような条件分岐作成機能、for 文と while 文のような繰り返し処理機能、任意の秒数だけ待つ機能およびプログラムを終了する機能がある。

これらの機能から、3.3.1 項で述べた訓練要素を実現するにあたり必要となる機能を選定した。表 3.4 に、プログラムブロックに採用した機能と採用しなかった機能を示す。

表 3.3: Scratch の基本的な機能群一覧

機能群名	機能群の説明
動き	オブジェクトを動かす機能
見た目	オブジェクトの見た目を変化させる機能
音	オブジェクトの動きに合わせて音を出したりする機能
特定変数	時間やタイマー、オブジェクトの座標など特定の変数を扱う機能
データ	変数や配列を作成する機能
演算子	四則演算や、数字の大小を調べる機能
特定条件	予め決められた条件を使って分岐を設定する機能
制御	条件分岐、待機および終了などのプログラムの制御機能

以下、これらの選定理由を説明する。

3.3.1 項で述べた訓練要素を実現するためには、訓練開始から終了まで常に火災や漏水などが発生し続けるのではなく、時間の経過や訓練体験者の行動内容に応じて任意のタイミングで発生および停止させる必要がある。そのため、「動き」に含まれる、ARで重畳表示するCGを再生する、停止するという2つの機能として「再生機能」および「停止機能」を採用した。また、表3.1の(1-h)、(2-c)および(2-h)のように他の訓練体験者、侵入者および警察官などの人物が移動する様子を模擬するために、人物CGの位置を変化させる必要がある。その際、人物CGが向いている方向と移動する方向が一致していなければ不自然になってしまう。そのため、プログラムブロックの配置に応じて、人物CGの3次元座標を変化させる機能および向きを変更する機能を「姿勢変更機能」として採用した。

「見た目」の機能群に相当する機能に関しては、人物CGなどの表示タイミングを任意に設定できるように、CGの表示および非表示を切り替える機能として「表示切り替え機能」を採用した。一方、火のCGの色を緑色に変更したり、人物CGの大きさを3メートルに変更したりする機能は、ゲームを作成する際には使用されることが多いが、現実的な訓練を作成する際には不要である。そのため、CGの色や大きさを変更する機能である「外見変更機能」は採用しなかった。表3.1の(1-d)時間経過により火が燃え広がる様子を模擬するには、図3.8のように、複数の火のCGを並べて設定することによって比較的容易に模擬できるため、採用しなかった。

「音」の機能群に相当する機能に関して、効果音は訓練の臨場感を向上させる重要な要素の1つだと考えられるため、CGの再生と同時に、そのCGに応じた効果音を再

生ずる機能として「CG音再生機能」を採用した。また、表3.1の(2-f)ユーザが通報する様子を模擬するためには、通報時の電話の音や警察の受け答えなどの音声ファイルを再生する必要がある。そのため、CGを伴わない特定の効果音を再生する機能として「効果音再生機能」を採用した。

「特定変数」の機能群に相当する機能に関して、表3.1の(1-b)時間経過によって火を発生や(1-h)人物が移動する様子を模擬するためには、訓練開始からの経過時間を取得し、それをを用いて条件分岐ができるようにする必要がある。そのため、訓練開始からの経過時間を取得する変数として「時間変数」を採用した。表3.1の(1-f)ユーザがある特定の位置に近づくことにより鎮火する様子を模擬するためには、ユーザとプログラムブロックが設置された位置の間の距離を取得し、それをを用いて条件分岐を設定する必要がある。そのため、ユーザとプログラムブロックが設置された位置の間の距離を取得する変数として「距離変数」を採用した。表3.1の(2-h)通報を受け一定時間経過後に警察官が到着する様子を実現するためには、通報したというイベント発生からの経過時間を取得し、それをを用いて条件分岐ができるようにする必要がある。そのため、特定のイベント発生からの経過時間を取得する変数として「イベント時間変数」を採用した。

3.3.1項で述べた訓練要素は、上記の複数の「特定変数」で実現できると考えられるため、「データ」の任意の変数や配列を作成する機能である「変数作成機能」と「配列作成機能」は採用しなかった。

3.3.1項で述べた訓練要素を実現するためには、条件分岐の条件として、「特定変数」を任意の数字と比較する必要がある。これは「演算子」の、数字の大小を調べる機能に相当する。そのため、「特定変数」を任意の数字と比較する機能として、「>」、「<」および「=」を採用した。一方、3.3.1項で述べた訓練要素を実現するにあたって、「特定変数」と任意の数字の間に四則演算する必要はないため、「四則演算」の機能は採用しなかった。

表3.1の、訓練開始時における漏水の発生を模擬するためには、「訓練の開始」を条件として分岐を設定する必要がある。「訓練の開始」は「特定条件」の、予め設定された条件に相当する。しかし、これは「時間変数」「=」「0」のように、「時間変数」を用いて実現できる。そのため、「特定条件」の機能は採用しなかった。

表3.1の、経過時間によって火の発生や人の移動などのイベント発生を模擬するためには、「時間変数」を任意の数字と比較し、任意の数字と同じであればイベントを発生させ、同じでなければ何もしない、という条件分岐を設定する必要がある。また、表

3.1の、ユーザが近づくことにより火が鎮火する様子を模擬するためには、「距離変数」を任意の数字と比較する条件を作成し、その条件を満たすならば火を発生させる、満たさないならば何もしない、という条件分岐を設定する必要がある。このように、「特定変数」を任意の数字と比較し、その真偽によってシナリオの進行が変化するような条件分岐を設定する機能が必要である。これはプログラミング言語におけるif文に相当する機能である。この、条件分岐を設定する機能として「if文」を採用した。一方、3.3.1項で述べた訓練要素を実現するにあたって、繰り返し機能と任意の秒数だけ待つ機能は必要ないため、これらは採用しなかった。訓練を終了する機能は、プログラムブロックを用いなくてもシステムを終了させればよいため、採用しなかった。

表 3.4: プログラムブロックに採用した機能と採用しなかった機能

機能群名	採用した機能	採用しなかった機能
動き	再生機能 停止機能 姿勢変更機能	
見た目	表示切り替え機能	外見変更機能
音	音声再生機能 効果音再生機能	
特定変数	時間変数 距離変数 イベント時間変数	
データ		変数作成機能 配列作成機能
演算子	> < =	四則演算
特定条件		訓練の開始
制御	if文	任意の秒だけ待つ機能 繰り返し機能 終了機能



### 3.3.3 プログラムブロックの形状の検討

一般的なプログラミング言語では、数値のみを代入できる変数に文字列を代入しようとするなど、間違っただ記述をした場合、エラーが表示されユーザは間違いに気付くことができる。一方、ビジュアルプログラミングでは、記述のルール違反を犯せないように設計されている場合が多く、ユーザの間違いを未然に防ぐことができる。実世界でプログラミングを行うプログラムブロックにおいて、ビジュアルプログラミングのようにユーザの間違いを未然に防ぐために、間違っただ組み合わせではブロック同士を組み合わせられない形状にするなどの物理的な工夫をすることが考えられる。また、プログラミングの高度なスキルを持っていない人であっても容易に理解できるように、どのブロック同士を組み合わせられるかどうかをユーザが直観的に理解できる形状にすることも考えられる。以上を考慮した上で、3.3.2項で述べたプログラムブロックの機能を基に、プログラムブロックの形状を検討した。以下、その詳細を述べる。

訓練のシナリオは、時間経過やユーザの行動などのイベントの発生によって人や物の状態が変化していくことにより進行する。そのため、シナリオをプログラムブロックで実現するためには、イベントの発生と、人や物の状態の変化の2つを設定する必要がある。3.3.2項で述べたプログラムブロックの機能においては、イベントの発生を設定する機能は「制御」「特定変数」「演算子」の3つ、人や物の状態の変化を設定する機能は「動き」「見た目」「音」の3つである。そこで、図3.15に示すように、イベントの発生を設定する条件ブロックと、人や物の状態の変化を設定する動作ブロックの2種類のブロックを組み合わせることで、シナリオの一部を示す塊を作成することとした。以下、このシナリオの一部を示す塊をシナリオブロックと呼ぶ。

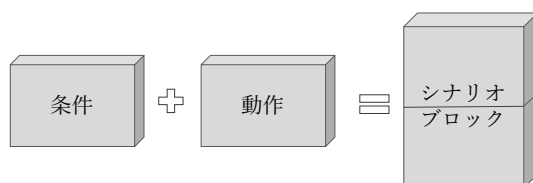


図 3.15: 条件と動作でシナリオの一部を作成する様子

条件ブロックでは、「if文」、「特定変数」および「演算子」を組み合わせることによってイベントの発生を設定する。例えば、「訓練開始から60秒経過したとき」というイベントの発生は、「もし『時間変数』が『60秒』『と同じ』であれば」と表記できる。文全体が「if文」に相当し、「と同じ」は「演算子」の「=」に相当する。

このように、条件ブロックでは、「もし『特定変数』が『数字』『演算子』ならば」のように「if文」というベースがあり、「特定変数」「数字」「演算子」というパーツをベースに組み合わせることによって、イベントの発生を設定できる。そこで図3.16に示すように、「if文」に相当する条件ベースブロックに、「特定変数」「数字」「演算子」に相当するパーツブロックを組み合わせることにより、条件ブロックを作成できるように設計した。

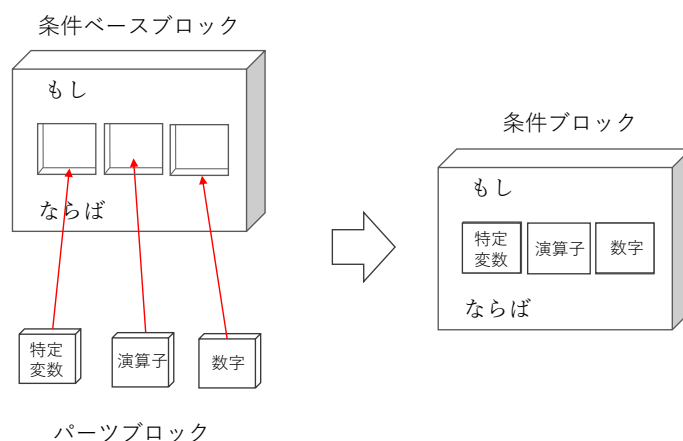


図 3.16: ベースブロックとパーツブロックで条件ブロックを作成する様子

動作ブロックでは、「オブジェクト」と「動き」を組み合わせることによって、人や物の状態の変化を設定する。例えば、「火および火の効果音」と「再生機能」を組み合わせ、火災の発生を模擬できる。また、「侵入者」と「姿勢変更機能」を組み合わせ、侵入者が移動する様子を模擬できる。このように人や物の状態の変化は、「オブジェクト」と「動き」を組み合わせることで設定できる。そこで図3.17に示すように、動作ベースブロックと呼ぶ枠に、「オブジェクト」「動き」というパーツブロックをはめ込むことにより、動作ブロックを作成できるように設計した。

図3.18に、シナリオブロックを作成する様子を示す。前述したように、ベースブロックとパーツブロックを組み合わせることで、条件ブロックと動作ブロックを作成する。ユーザがパーツブロックの形を見て、どのベースブロックのどの場所にはめ込めばよいかを直観的に理解できるように、パーツブロックの形状を条件設定用パーツブロックと動作設定用パーツブロックで異なるものとし、各ベースブロックの穴の形は、対応したパーツブロックのみを取り付けられるように作成した。また、ユーザが条件ベースブロックと動作ベースブロックを見て、2つを組み合わせる必要があることを直感的に理解できるように、条件ベースブロックの下部に穴を、動作ベースブロックの上部

に突起をつけ、条件ベースブロックの下部に動作ベースブロックを差し込む形状にした。また、火災の発生を模擬するために煙と火を同じ条件で同じ場所で発生させる場合のように、1つの条件ブロックに対して複数の動作ブロックを設定する場合を想定して、図 3.19 に示すように、動作ベースブロックを他の動作ベースブロックの下方向に組み合わせられるように設計した。

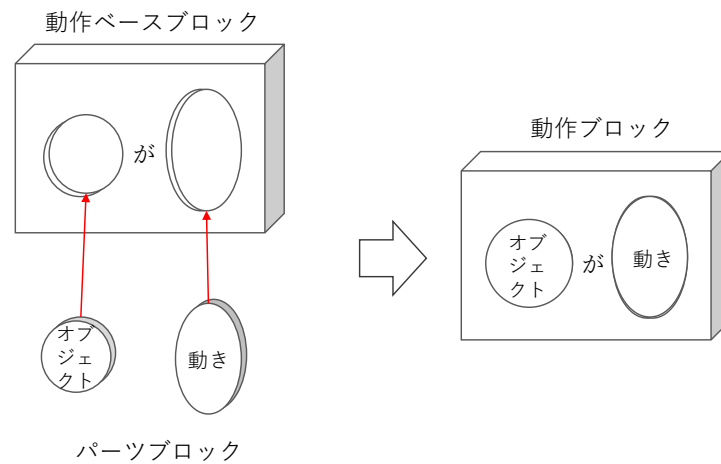


図 3.17: ベースブロックとパーツブロックで動作ブロックを作成する様子

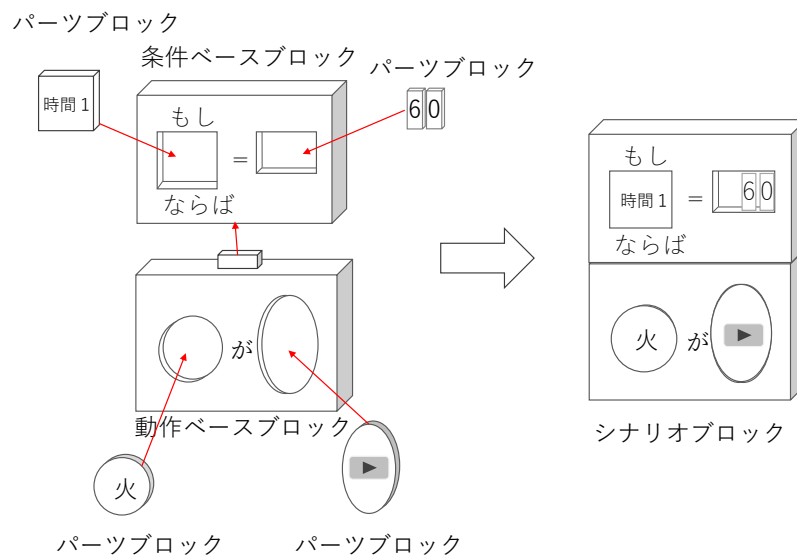


図 3.18: シナリオブロックを作成する様子

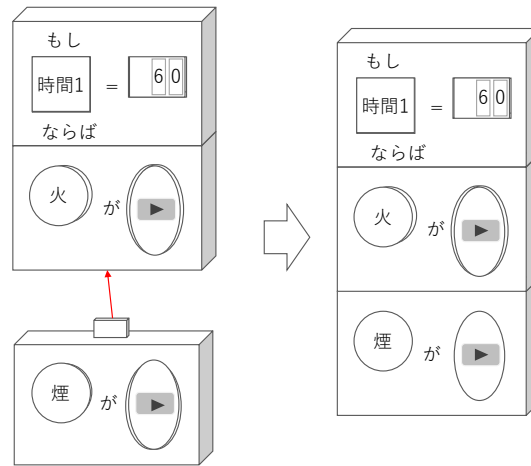


図 3.19: 複数の動作ブロックを組み合わせる様子

3.3.2 項で述べたプログラムブロックの機能を、ベースブロックとパーツブロックに当てはめて設計した。図 3.20 に示すように、「特定変数」の「距離変数」として距離パーツブロック、「時間変数」として時間1 パーツブロック、「イベント時間変数」として時間2、時間3、時間4、時間5 パーツブロックを設計した。

ユーザとプログラムブロックの距離の測定は、現状の技術では 1m 単位が正確に測定できる限界だと考えられる。理想的には cm 単位の精度で設定できることが望ましいが、1m 単位の設定でも多くの状況を表現できると考え、「距離変数」の単位は m とした。また、3.3.1 項で述べたプログラムブロックで実現する訓練要素を設定するためには、秒単位で条件分岐を設定する必要がある。そのため、「時間変数」「イベント時間変数」の単位は秒とした。ユーザが訓練環境を移動する範囲を考慮すると、距離は最大 50 m 程度が表現できれば十分だと考えられる。また、訓練実施時間は最大 2 時間程度であると考えられる。そのため、これらを表現できる数字の桁数として、最大 4 桁を設定できるようにした。

図 3.21 に示すように、任意の数字を設定するために用いる、0 から 9 の数字を示す数字パーツブロックを設計した。

また、条件ベースブロックは、上記の「特定変数」のパーツブロックと数字パーツブロックをはめ込む穴を持つ形状に設計した。また、「演算子」のパーツブロックは作成せず、図 3.22 に示すように、条件ベースブロックを「演算子」と組み合わせ、「>」、「<」および「=」の「演算子」をもつ 3 種類の条件ベースブロックを設計した。先に述べたように、最大 4 桁の数字パーツブロックをはめ込められるように、条件ベースブロックの穴を設計した。

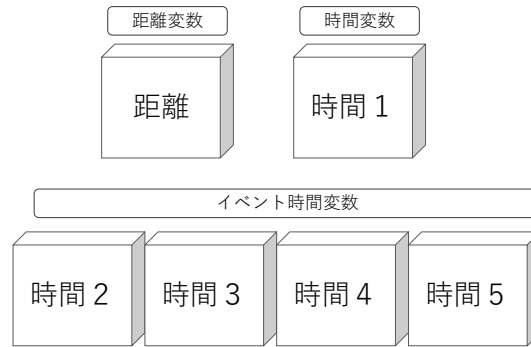


図 3.20: 「特定変数」のパーツブロックの種類

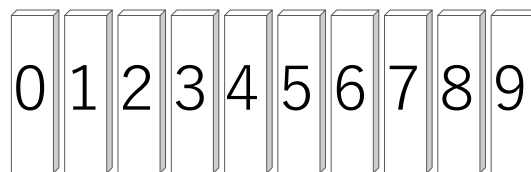


図 3.21: 数字のパーツブロックの種類

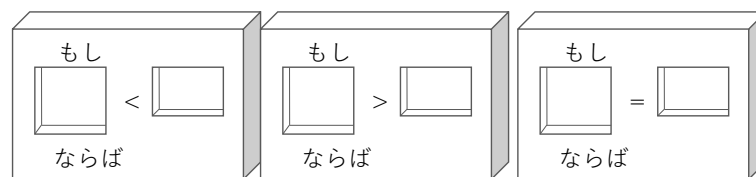


図 3.22: 条件ベースブロックの種類

図 3.23 に示すように、「オブジェクト」の「漏水および漏水の効果音」として漏水パーツブロック、「火および火の効果音」として火パーツブロック、「煙」として煙パーツブロック、「爆発および爆発の効果音」として爆発パーツブロック、「通報音声」として通報パーツブロック、「他の訓練体験者」として作業員パーツブロック、「侵入者」として侵入者パーツブロック、「警察官」として警察官パーツブロックを設計した。

また、図 3.24 に示すように、「再生機能」として再生パーツブロック、「停止機能」として停止パーツブロック、「姿勢変更機能」として動くパーツブロック、乗り越えるパーツブロック、飛び降りるパーツブロックおよびかがむパーツブロックを設計した。

また、例えば火パーツブロックとかがむパーツブロックを組み合わせることは間違いであるように、「オブジェクト」が人であるか物であるかによって、「動き」の部分

に設置可能なパーツブロックは異なるため、人と物のパーツブロックの形状は異なるものとした。

図 3.25 に示すように、動作ベースブロックは、物パーツブロック用のベースブロックと、人パーツブロック用のベースブロック、そして任意のイベントからの時間を開始する用のベースブロック（以下、時間ベースブロック）の 3 種類を設計した。例えば、時間ベースブロックに時間 2 パーツブロックをはめ込むことで、「『時間 2』を開始する」というイベントの発生を設定できる。その後は時間 2 パーツブロックを用いて、「『時間 2』を開始する」というイベントの発生からの経過時間を取得できる。

前述したようにパーツブロックの形状をその種類に応じて異なるものにする事で、図 3.26 に示すように、ユーザがパーツブロックをベースブロックの間違った箇所にはめ込めないように設計し、ユーザのミス未然に防ぐことができるようにした。

図 3.27 に、ARTM で採用するプログラムブロック一覧を示す。また、これらのプログラムブロックを用いて作成可能なシナリオの例を付録 A の図 A.1 から図 A.7 に示す。



図 3.23: オブジェクトパーツブロックの種類

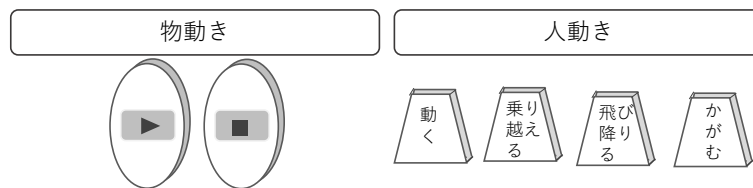


図 3.24: 「動き」のパーツブロックの種類

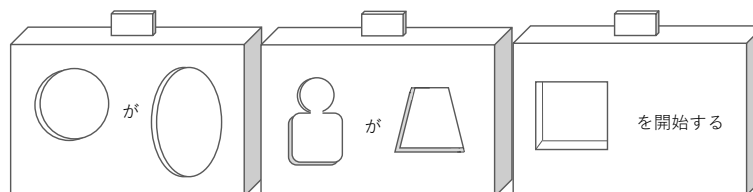


図 3.25: 動作ベースブロックの種類

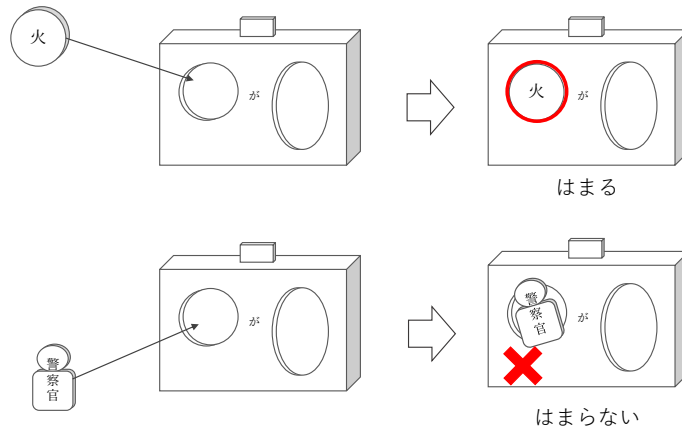


図 3.26: パーツブロックをベースブロックの間違った箇所にはめ込めない様子

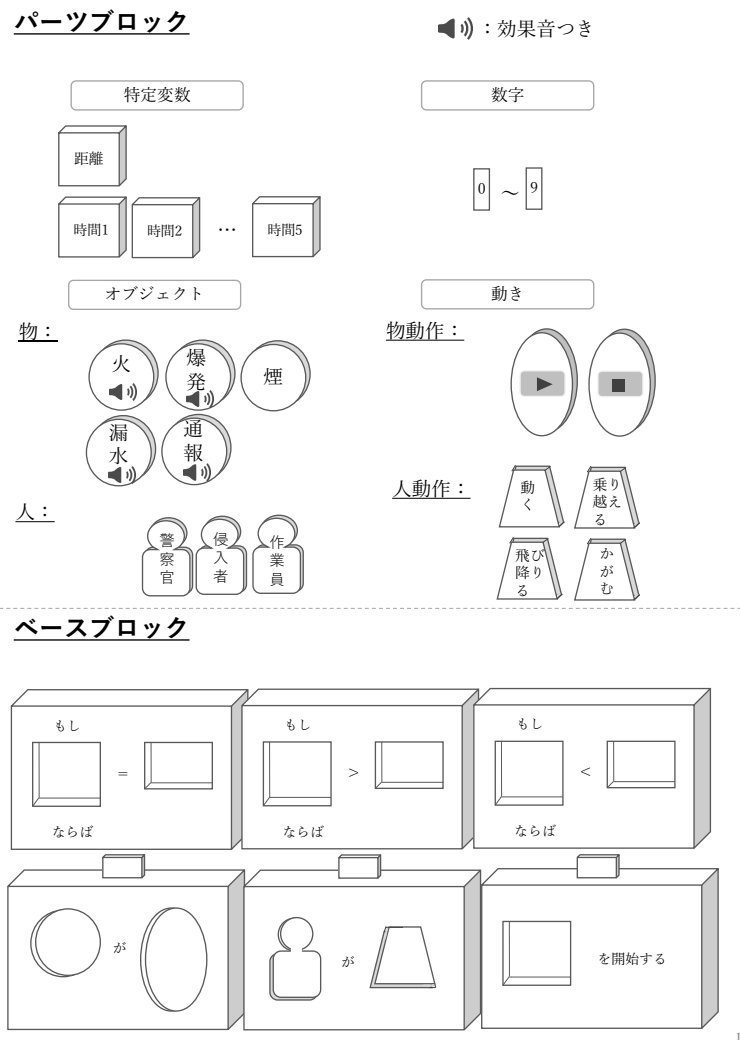


図 3.27: プログラムブロック一覧

### 3.4 訓練環境構築サブシステムの設計

本節では ARTM を構成するサブシステムの 1 つである訓練環境構築サブシステムの設計について述べる。図 3.28 に訓練環境構築サブシステムの使用時の流れを示す。はじめに、訓練環境構築者はプログラムブロックを組み立て、訓練シナリオの元となる複数のシナリオブロックを作成する。その後、作成したシナリオブロックを訓練環境中に配置する。最後にタブレット PC の画面を見ながら、シナリオブロックと訓練環境全体を撮影することで、シナリオブロックの内容や位置、向きおよび訓練環境体験時のカメラ姿勢の推定に必要なデータを取得する。

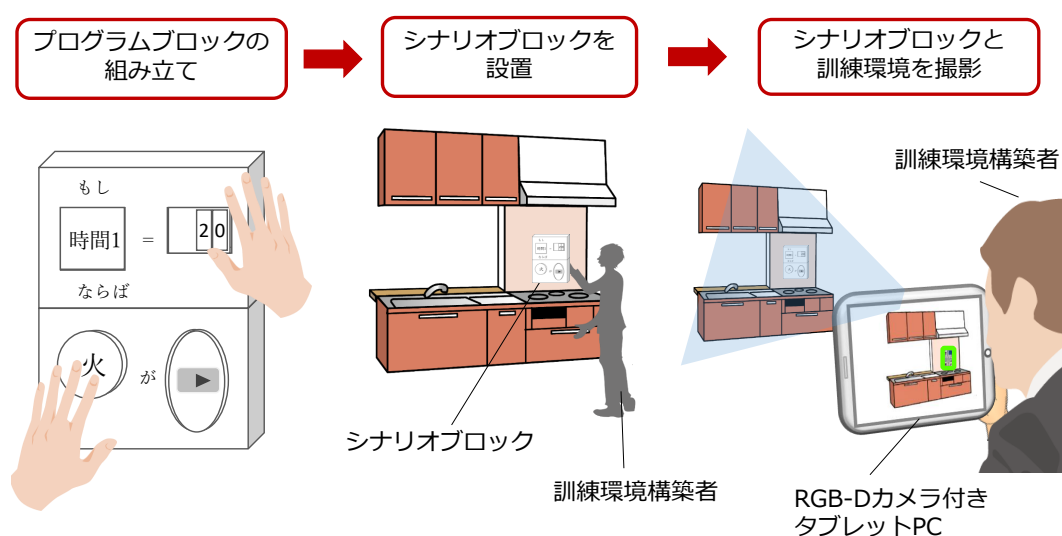


図 3.28: 訓練環境構築サブシステムの使用時の流れ

以下ではシナリオブロックの認識に必要なカメラ姿勢の推定方法の検討および、その際に用いる表示デバイスとしてタブレット PC を採用した理由について述べる。その後、訓練環境構築サブシステムの設計の詳細について述べる。

#### 3.4.1 トラッキング方法と表示デバイスの検討

訓練環境構築者が配置したシナリオブロックを、システムが認識し、シナリオを取り込むためには、配置した各シナリオブロックがそれぞれの位置かつどの向きで配置されているか（以下、シナリオブロックの姿勢）および、配置した各シナリオブロックがそれぞれのどのようなベースブロックおよびパーツブロックで構成されているか（以下、シナリオブロックの内容）を認識する必要がある。本サブシステムでは、ユーザが撮影したカメラ画像群からシナリオブロックを認識する。1 枚の画像から、配置したす



すべてのシナリオブロックの姿勢および内容を認識することは難しいため、ユーザは訓練環境を歩き回り、配置したすべてのシナリオブロックをある程度近い距離から撮影する必要がある。撮影によって得られた画像群から、シナリオブロックの姿勢を認識するためには、各画像を取得した時のカメラの位置および向きを推定する（以下、トラッキング）必要がある。

また、配置したすべてのシナリオブロックを抜け落ちなく撮影するためには、どのシナリオブロックが撮影済みで、どのシナリオブロックが未撮影なのかをユーザが把握できる必要がある。そのため、シナリオブロックの認識状況をユーザに提示するための表示デバイスが必要である。

以上のことから、本サブシステムに適したトラッキング方法および表示デバイスを検討した。以下に、その詳細を述べる。

#### (1) トラッキング方法の検討

トラッキング方法は大きく分けて、慣性センサ法、GPS法、磁気センサ法、人工マーカ法、マーカレス法に分類される<sup>[26]</sup>。

慣性センサ法では、加速度センサとジャイロセンサを用いる。加速度センサによるユーザの速度の計測結果と、ジャイロセンサによるユーザの向きの計測結果を用いて、ユーザの単位時間あたりの姿勢の変化を追跡し、ユーザの姿勢を推定する。しかし慣性センサ法は、時間経過に伴ってトラッキング結果に誤差が蓄積するという精度面の問題点があるため、長時間使用する必要があると想定される本サブシステムには適していない。

GPS法は、GPS衛星から発信される電波を利用してセンサの緯度経度情報を計測する方法である<sup>[27]</sup>。GPSからの電波を受信できない屋内では使用できないため、GPS法は屋内での使用を想定した本サブシステムには適していない。

磁気センサ法では、トランスミッタから発生する磁気を磁気センサで検出することによって、センサの姿勢を推定する<sup>[28]</sup>。しかし、磁気センサで計測できる範囲は比較的狭く、広い施設内を訓練環境とする本サブシステムには適していない。

人工マーカ法では、事前準備として、3次元位置情報を計測した人工マーカを環境内に配置する。このマーカをカメラで撮影し、得られた画像におけるマーカの2次元位置とマーカの実空間における3次元位置情報を用いて、カメラの姿勢を推定する手法である<sup>[26]</sup>。人工マーカ法は他のトラッキング方法と比べると精度と安定性が高い方法であるが、予め人工マーカを設置しその3次元位置を計測する手間が必要であるため、本サブシステムには適していない。

マーカレス法は、画像内の直線や角など環境に予め存在する画像認識で識別可能な特徴を用いる方法である。カメラで撮影した画像内の3次元位置が既知の自然特徴とカメラ間の相対的な位置関係を計算することによって、カメラの姿勢を推定する<sup>[29]</sup>。この方法は、環境中に特別な機器を設置する必要がなく、広範囲で使用できるという利点がある。また、自然特徴が大きく変化する環境では使用できないという欠点があるが、訓練環境は比較的静的な環境であると考えられるため、マーカレス法は本サブシステムに適していると考えられる。

よって本サブシステムでは、マーカレス法の1つである、深度情報も取得できるRGB-Dカメラを用いたマーカレストラッキングを採用した。マーカレストラッキングの詳細については付録Bで述べる。

## (2) 表示デバイスの検討

本サブシステムではRGB-Dカメラを用いたマーカレストラッキングを用いるため、訓練環境をRGB-Dカメラを用いて撮影する。撮影した画像を用いてプログラムブロックを認識する際、プログラムブロックの認識に失敗することがある。そのため、どのプログラムブロックは認識できているかという認識状況を訓練環境構築者に提示する必要がある。そこで、認識状況を提示することに適した表示デバイスを検討した。

ユーザが広い範囲を動くことを想定した場合、表示デバイスは以下の3種類が考えられる。

- ヘッドマウント型ディスプレイ (HMD)
- プロジェクション型ディスプレイ
- ハンドヘルド型ディスプレイ

HMDは、図3.29に示すような、ユーザが頭に装着するタイプのディスプレイである。内蔵しているカメラや深度センサを用いて、AR表示を可能にできる。HMDには、ユーザが両手を自由に使用できるという利点がある。また、ユーザの視界の多くをディスプレイが占有するため、ユーザは高い没入感を得られる。HMDには、ビデオシースルー型HMDと光学シースルー型HMDの2種類がある。ビデオシースルー型HMDは一般的に周辺視野が遮蔽されるため、装着しながら様々な機器が設置された訓練環境中を動き回るとは危険である。光学シースルー型HMDはユーザの視野は制限しないが、重畳位置が不正確である場合があるため、どのプログラムブロックを認識したか正しく提示できない可能性がある。以上の理由からHMDは本サブシステムには適していない。



図 3.29: ヘッドマウント型ディスプレイの例 [23]

プロジェクション型ディスプレイは、プロジェクタを用いてCGを実環境に直接投影する方法である。このようなディスプレイには、プロジェクタを環境中に設置するものと、図3.30に例を示すように、ユーザが装着するものの2種類が存在する。このようなディスプレイは、実環境に直接CGを投影するため、多人数でCGを参照できるという利点がある。しかしプロジェクション型ディスプレイは明るい環境で使用することは難しいため、本サブシステムには適していない。



図 3.30: プロジェクション型ディスプレイの例 [24]

ハンドヘルド型ディスプレイは、図3.31に示すように、ユーザが手をもって使用するタイプのディスプレイである。ハンドヘルド型ディスプレイは携帯性に優れており、ユーザの視界を妨げることがないという利点がある。そのため、ユーザが訓練環境中を動き回ることが想定される本サブシステムにおいても安全かつ簡便に使用できるため、ハンドヘルド型ディスプレイは適していると考えられる。

ハンドヘルド型ディスプレイには、スマートフォンやタブレット PC など、様々な大きさのものがある。スマートフォンは軽量で携帯性に優れるが、画面が小さいという欠点がある。画面が小さいと、ユーザがシナリオブロックの認識状況を確認しづらくなるため、本サブシステムには適していない。タブレット PC は画面が大きく AR 表示を確認しやすい。しかし大きいほど、重量も大きくなりユーザの負担になる。

以上の理由から、本研究では表示デバイスとして、ユーザの負担になりにくい 1kg 程度の適度な大きさのタブレット PC を採用した。



図 3.31: ハンドヘルド型ディスプレイの例 [25]

### 3.4.2 訓練環境構築サブシステムの詳細設計

図 3.32 に訓練環境構築サブシステムのシステム構成を示す。3.4.1 項で述べたように、本システムでは表示デバイスとしてタブレット PC を採用した。また、マーカレストラッキングを行うため、RGB-D カメラをタブレット PC に取り付けた。タブレット PC ではこの取り付けた RGB-D カメラから color 画像と depth 画像を取得する。

マーカレストラッキングでは、環境を撮影して得られた color 画像と depth 画像を用いてトラッキングする。しかし、マーカレストラッキングは CPU への負荷が高いため、タブレット PC の性能ではリアルタイムでのトラッキングが困難である。そのため、本サブシステムでは、color 画像と depth 画像を別の高性能 PC に無線通信を用いて転送し、高性能 PC でトラッキングやシナリオブロックを認識し、シナリオブロックの認識

状況を示した合成画像を再び無線通信を用いてタブレット PC に返送して表示する方式を用いた。高性能 PC 上でトラッキングおよびタブレット PC との通信を実行するアプリケーションを構築サーバアプリ、タブレット PC 上で画像の取得および高性能 PC との通信を実行するアプリケーションをクライアントアプリと呼ぶ。

本サブシステムは最終的に、AR 訓練体験時に必要なデータであるシナリオブロックの位置、向き、内容および、トラッキングに関するデータ（カメラ内部のパラメータ、リローカリゼーションに必要な情報、キーフレーム情報、キーフレーム間の関係情報）を取得する。このシナリオブロックの位置、向きおよび内容を記したファイルをシナリオファイルと呼ぶ。また、トラッキングに関するデータをトラッキングデータと呼ぶ。

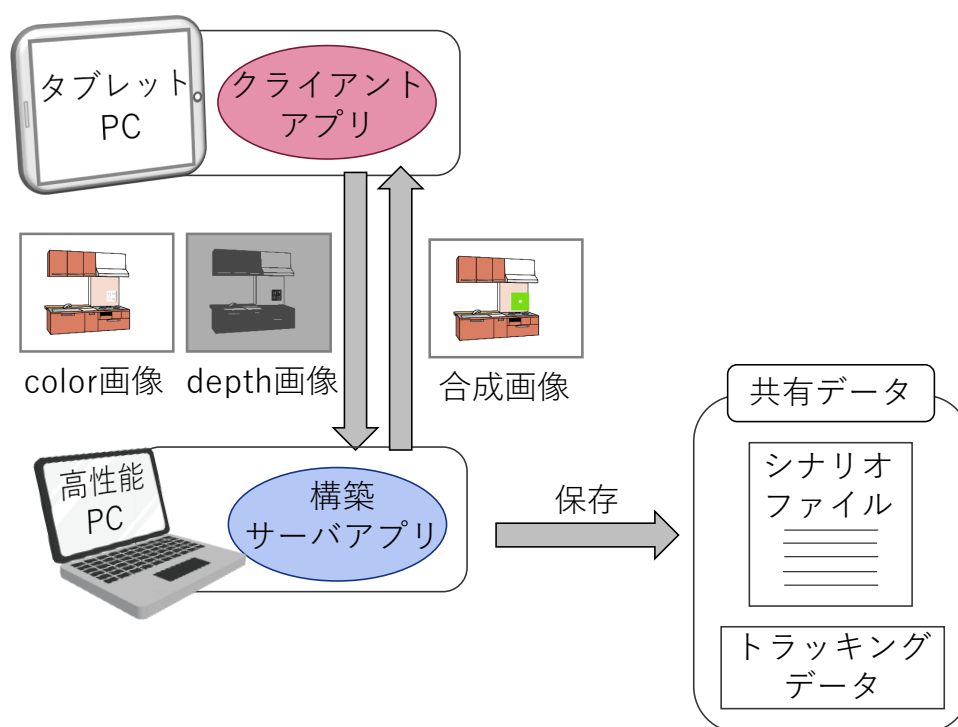


図 3.32: 訓練環境構築サブシステムのシステム構成

図 3.33 にクライアントアプリの処理の流れを示す。クライアントアプリでは、はじめに構築サーバアプリとの通信を開始する。通信確立後、RGB-D カメラから color 画像と depth 画像を取得し、それらを無線通信で構築サーバアプリに送信する。その後、構築サーバアプリから、シナリオブロックの認識状況を示した合成画像を受信し、それを画面上に表示する。通信接続が切れるまでこれを繰り返す。

図 3.34 に構築サーバアプリの処理の流れを示す。はじめに、構築サーバアプリはクライアントアプリと通信を開始する。その後、クライアントアプリから受信した color

画像と depth 画像を基にトラッキングする。トラッキング失敗時は、図 3.35 に示すように、トラッキングに失敗していることをユーザに伝えるためのテキストを color 画像上に表示した合成画像をタブレット PC のディスプレイに表示し、ユーザが確認できるようにした。トラッキング成功時は、color 画像内のシナリオブロックを認識する。color 画像内に既に認識したシナリオブロックがある場合、図 3.36 に示すように、緑色のハイライトを表示してユーザが確認できるようにした。認識しなかった場合は何も表示されない。画像にプログラムブロックが写っているときにブロックを認識する具体的な処理の流れは 3.6.1 項で述べる。環境中に設置したすべてのシナリオブロックを撮影し終わったら、高性能 PC の終了キーを押すことで撮影を終了する。撮影終了後、トラッキングデータとシナリオファイルが保存される。

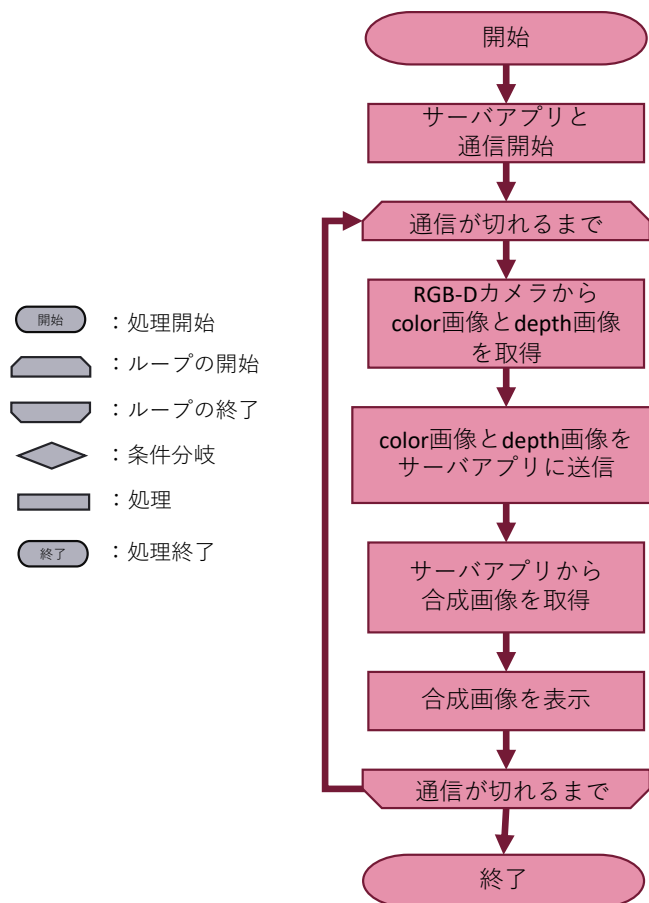


図 3.33: クライアントアプリの処理の流れ

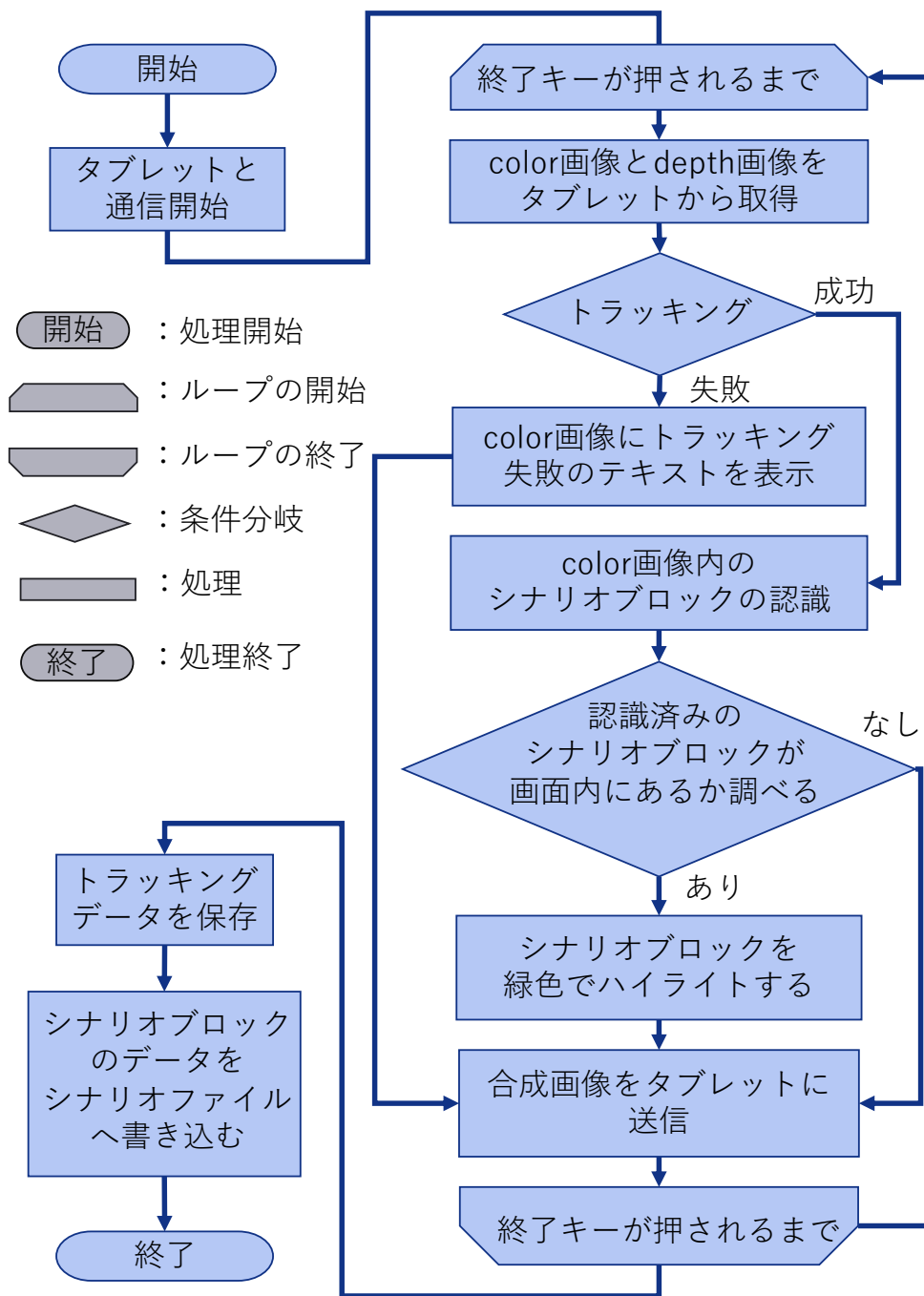


図 3.34: 構築サーバアプリの処理の流れ

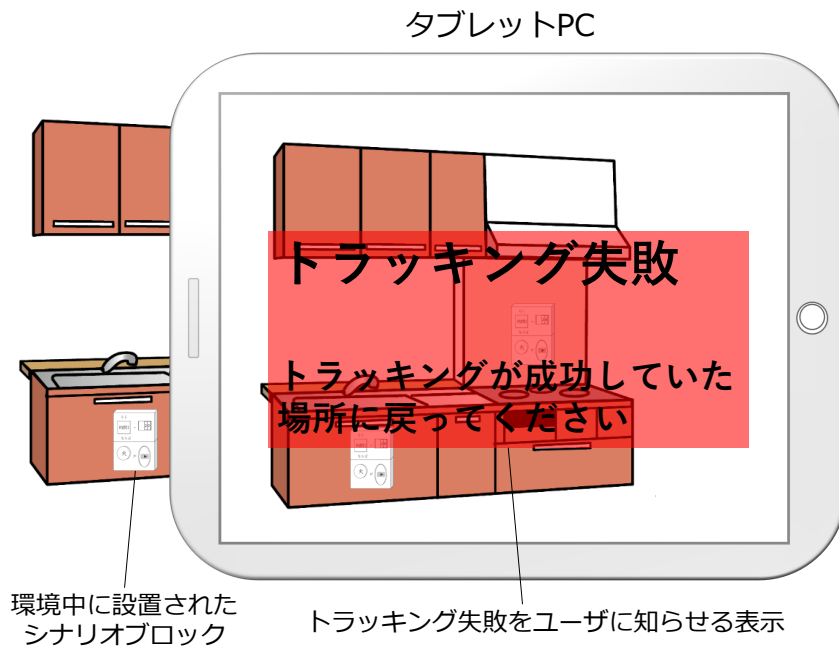


図 3.35: トラッキング失敗を示した合成画像を表示するイメージ

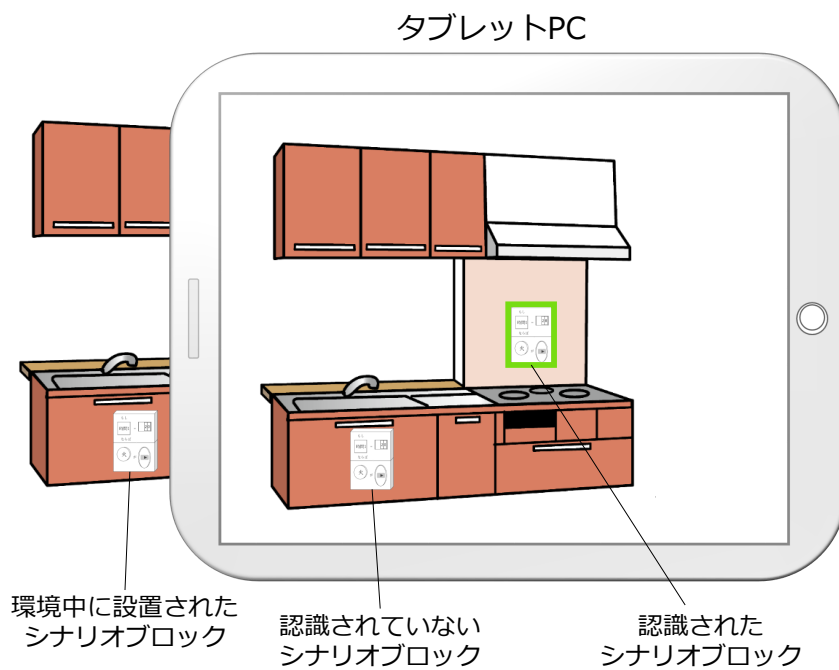


図 3.36: 認識したシナリオブロック上に緑色のハイライトを表示するイメージ



### 3.5 訓練環境体験サブシステムの設計

本節では ARTM を構成するサブシステムの 1 つである訓練環境体験サブシステムの設計について述べる。図 3.37 に訓練環境体験サブシステムの使用時のイメージを示す。訓練体験者はタブレット PC を通して訓練環境を見ることで、訓練環境中に設置されたシナリオブロックの位置や向きおよび内容に応じて重畳表示された CG を確認できる。また、シナリオに応じて環境に置かれた物から発生する効果音も聞くことができる。

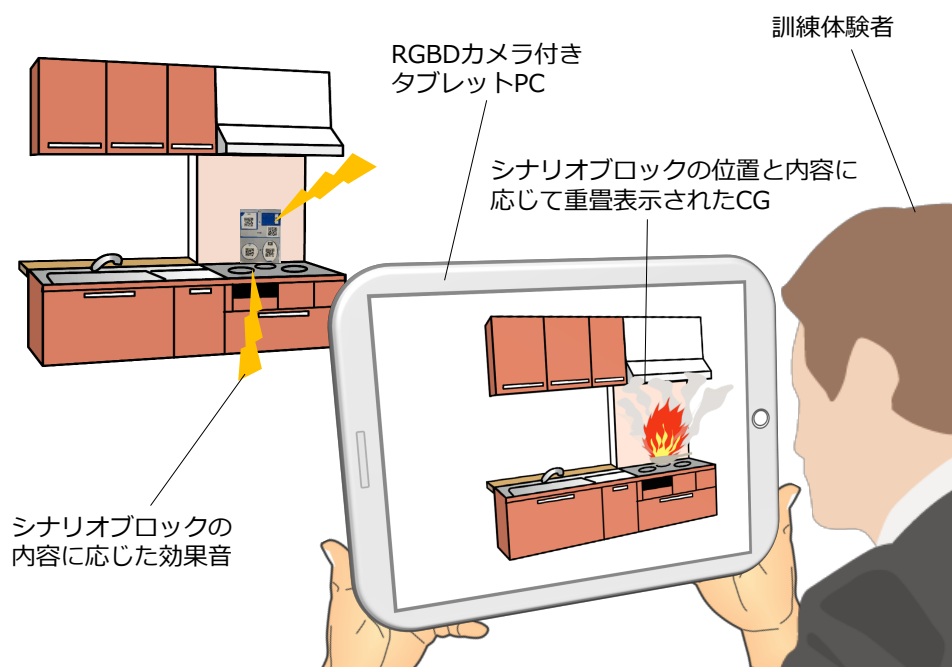


図 3.37: 訓練環境体験サブシステムの使用時のイメージ

#### 3.5.1 トラッキング方法と表示デバイスの検討

本サブシステムでは、AR 表示する際の CG の位置合わせのために、トラッキングする必要がある。また、AR 表示をユーザが確認するための表示デバイスも必要である。以上のことから、本サブシステムに適したトラッキング方法と表示デバイスを検討した。以下に、その詳細を述べる。

##### (1) トラッキング方法の検討

トラッキング方法は訓練環境構築サブシステムと同様の理由から、RGB-D カメラを用いたマーカレストラッキングを採用した。

##### (2) 表示デバイスの検討

本サブシステムではユーザが AR 訓練環境を体験する際に、CG を確認できる表示デバイスが必要である。そのため、本サブシステムに適した表示デバイスを検討し、決定した。

ビデオシースルー型 HMD は、訓練環境構築サブシステムと同様に安全性の理由により、本サブシステムには適していない。光学シースルー型 HMD はユーザの視野は制限しないが、装着するたびにキャリブレーションする必要があるため、様々な人々に簡便に使用してもらうことを意図した本サブシステムにおいては効率的でない。以上の理由から HMD は本サブシステムには適していない。

プロジェクション型ディスプレイは訓練環境構築サブシステムと同様に明るい環境で使用できないため、本サブシステムには適していない。

ハンドヘルド型ディスプレイは、訓練環境構築サブシステムと同様の理由で、本サブシステムにも適していると考えられる。

以上の理由から、本サブシステムでは表示デバイスとして、タブレット PC を採用した。

### 3.5.2 訓練環境体験サブシステムの詳細設計

図 3.38 に、訓練環境体験サブシステムのシステム構成を示す。訓練環境体験サブシステムでは、マーカレストラッキングと並行して CG をレンダリングする必要があるため、CPU への負荷が高い。そのため、訓練環境構築サブシステムと同様に、タブレット PC と高性能 PC 間で無線通信を用いて画像の送受信を行う方式を用いた。高性能 PC でトラッキング、CG のレンダリングおよびタブレット PC との通信を行うアプリケーションを体験サーバアプリと呼ぶ。タブレット PC では訓練環境構築サブシステムと同様にクライアントアプリを用いた。また本サブシステムでは、効果音を発生させるため、一部のプログラムブロックにスピーカを取り付けた。高性能 PC とプログラムブロックに取り付けたスピーカは距離が離れているため、無線接続し、シナリオに応じて高性能 PC 内の音声データを無線接続したスピーカで再生する方式を用いた。

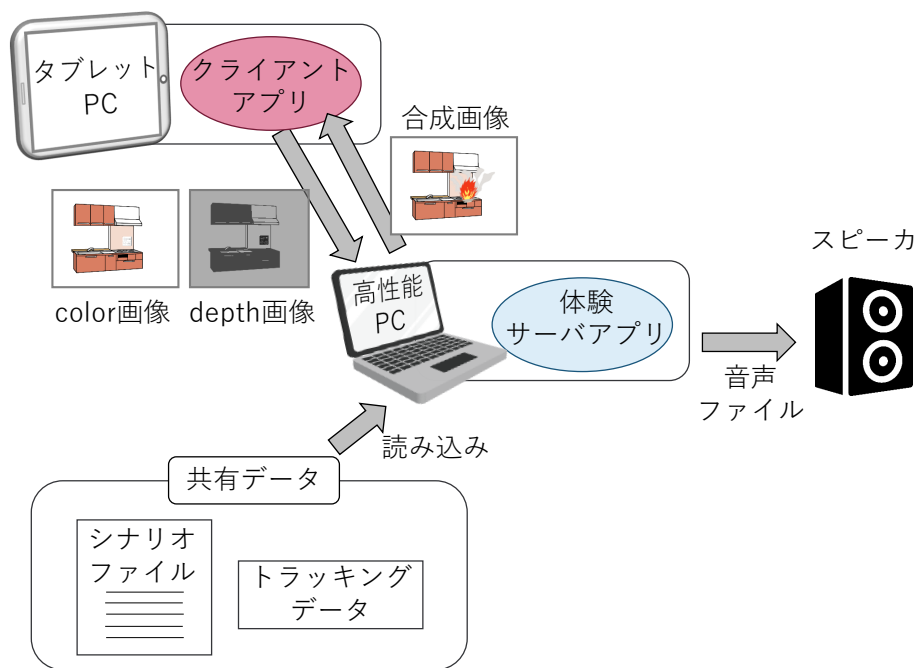


図 3.38: 訓練環境体験サブシステムのシステム構成

図 3.39 に体験サーバアプリの処理の流れを示す。体験サーバアプリでは、はじめに訓練環境構築サブシステムで保存したシナリオファイルとトラッキングデータを読み込む。シナリオファイルに記された情報を以下ではシナリオと呼ぶ。メインループに入る前に、訓練開始からの経過時間を表す変数「時間 1」を開始する。メインループのはじめでは、クライアントアプリから color 画像と depth 画像を取得する。その後、「時間 1」および「時間 2-5」と、シナリオ内の時間条件を比較する。時間条件を満たすものがあれば、その対となる動作を発生させる。その際、効果音が設定されていれば、スピーカから流す。その後、クライアントアプリから取得した color 画像と depth 画像を基にトラッキングを実行する。トラッキングに失敗した場合、訓練環境構築サブシステムと同様に、トラッキングに失敗していることをユーザに伝えるためのテキストを color 画像上に表示した合成画像をクライアントアプリに送信する。トラッキングに成功した場合は、ユーザとの距離を表す変数「距離」とシナリオ内の距離条件と比較する。距離条件を満たすものがあれば、その対となる動作を発生させる。時間イベントと同様に、効果音が設定されていれば、スピーカから流す。高性能 PC の終了キーを押すことで本サブシステムを終了する。

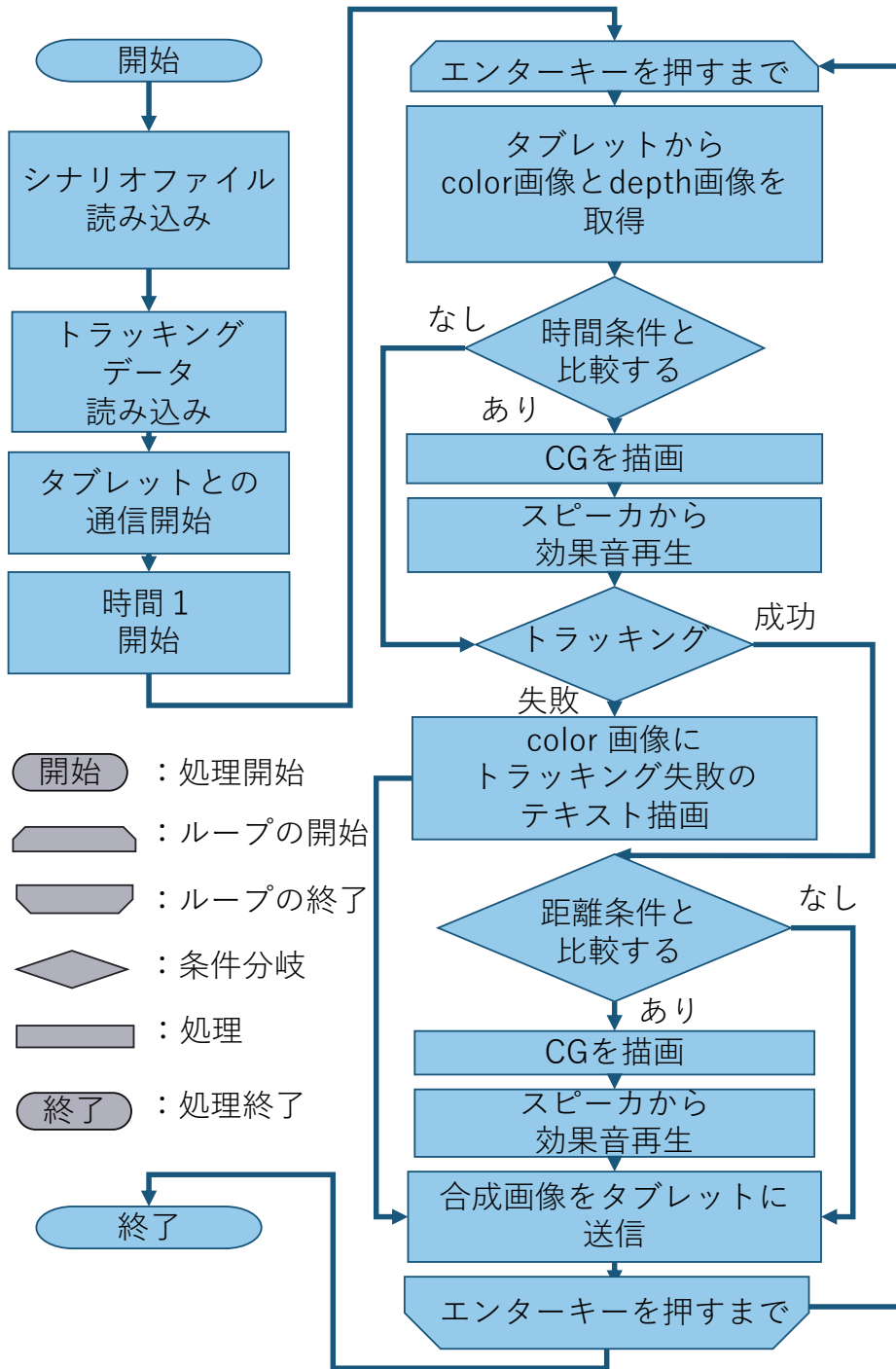


図 3.39: 体験サーバアプリの処理の流れ

## 3.6 ARTMの実装

### 3.6.1 訓練環境構築サブシステムの実装

本項では、3.4節で述べた設計に基づく訓練環境構築サブシステムの実装について述べる。図3.40に訓練環境構築サブシステムのハードウェア構成を示す。本サブシステムは、高性能ノートPC、タブレットPC、RGB-Dカメラおよび、高性能ノートPCとタブレットPCの無線接続に用いるルータで構成する。RGB-DカメラにはASUS XTion Pro Liveを用いた。本サブシステムで用いた高性能ノートPCの外観、タブレットPCの外観、RGB-Dカメラの外観を、図3.41から図3.43に、それぞれの仕様およびルータの仕様を、表3.5から表3.8に示す。

3.4.2項で述べたように、訓練環境構築サブシステムでは、タブレットPC上でクライアントアプリ、高性能ノートPC上で構築サーバアプリを実行する。

クライアントアプリおよび構築サーバアプリはMicrosoft Visual Studio 2017を用いて開発し、開発言語はC++を用いた。クライアントアプリと構築サーバアプリ間でのcolor画像とdepth画像、合成画像の通信にはboost1.69.0-1<sup>[30]</sup>のasioを用いた。クライアントアプリにおけるRGB-Dカメラからのcolor画像とdepth画像の取得にはOpenNI2<sup>[31]</sup>を用いた。その際、color画像およびdepth画像の解像度は640×480ピクセル、フレームレートは30fpsであった。構築サーバアプリにおける合成画像の生成にはOpenCV3.4.3<sup>[32]</sup>を用いた。



図 3.40: 訓練環境構築サブシステムのハードウェア構成



図 3.41: 本システムで用いた高性能ノート PC の外観

表 3.5: 本システムで用いた高性能 PC の仕様

OS	Windows 10 Pro 64bit
CPU	Intel(R)Core(TM) i7-7700HQ(2.80GHz)
GPU	NVIDIA GeForce GTX 1060
RAM	16GB
無線の種類	802.11ac
有線の種類	1000BASE-T



図 3.42: 本システムで用いたタブレット PC の外観

表 3.6: 本システムで用いたタブレット PC の仕様

OS	Windows 10 Home 64bit
CPU	Intel(R) Atom x5-Z8350 (1.44GHz)
RAM	4GB
無線の種類	802.11ac
大きさ	縦 17.5 × 横 26.0 × 幅 0.9cm
画面サイズ	10.1inch
重さ	1.08kg



図 3.43: 本システムで用いた RGB-D カメラの外観

表 3.7: 本システムで用いた RGB-D カメラの仕様

製品名	ASUS XTion Pro Live
color 画像の解像度 およびフレームレート	最大 SXGA 1280 × 1024 ピクセル (30fps) HD 1280 × 720 ピクセル (30fps) VGA 640 × 480 ピクセル (30fps) QVGA 320 × 240 ピクセル (60fps) QVGA 320 × 240 ピクセル (30fps)
depth 画像の解像度 およびフレームレート	最大 VGA 640 × 480 ピクセル (30fps) QVGA 320 × 240 ピクセル (60fps) QVGA 320 × 240 ピクセル (30fps) QQVGA 160 × 120 ピクセル (30fps)
深度計測手法	Structured Light
深度測定範囲	0.8~3.5m
FOV	水平 58 度, 垂直 45 度, 対角 70 度
大きさ	縦 3.5 × 横 18.0 × 幅 5.0cm
重さ	約 160g

表 3.8: 本システムで用いたルータの仕様

製品名	Buffalo WSR-A2533DHP2-CB
無線 LAN 通信規格	IEEE 802.11ac(約 433.3Mbps)
有線 LAN 通信規格	イーサネット (約 1.0Gbps)

以下では、構築サーバアプリにおける、シナリオブロックの認識処理の実装について述べる。

本サブシステムでは、ユーザはプログラムブロックを見ながら組み立てるため、ユーザがプログラムブロックの種類を目視で識別できる必要がある。そのため、図 3.27 に示したように、各プログラムブロックの種類を表記し、効果音を伴うパーツブロックには音声マークを表記した。本システムではユーザが撮影した画像からシナリオブロックの内容を認識するが、各プログラムブロックに表記された日本語の文字の内容を、システムが正確かつ安定して認識することは難しい。そのため、ユーザが撮影した画像からプログラムブロックの種類をシステムが正確かつ安定して識別するための仕組みを別途設ける必要がある。

また、1枚の画像内に複数のプログラムブロックが同時に映る場合もあると予想されるが、システムが画像内の各プログラムブロックを認識するにあたっては、どこまでが1つのシナリオブロックであるのかを認識する必要もある。3.3.3項で述べたように、シナリオブロックは条件ブロックと動作ブロックで構成されており、条件ブロックにはめられるパーツブロックおよび動作ブロックにはめられるパーツブロックは予め決まっているため、1つのシナリオブロック内での起こりうるプログラムブロックの組み合わせも予め決まっている。そこで、画面内に映るパーツブロックの組み合わせと位置関係を用いて、画面内に映った複数のシナリオブロックから個々のシナリオブロックを分けて認識することとした。

また、本システムではユーザが撮影した画像からシナリオブロックの内容だけでなく、姿勢も認識する必要がある。しかし、画像に写ったシナリオブロックの輪郭のみからその姿勢を認識することは難しい。そのため、ユーザが撮影した画像からシナリオブロックの姿勢を認識するための仕組みを別途設ける必要がある。

そこで、本システムではプログラムブロックの種類、シナリオブロックの認識およびシナリオブロックの姿勢の認識に、図 3.44 に示すような、ArUco マーカを利用した。ArUco<sup>[33][34]</sup> は OpenCV をベースにした AR ライブラリで、ArUco マーカはカメラの姿勢推定に使用できる正方形マーカである。ArUco マーカは内部にその ID 番



号を表すバイナリデータ領域を持ち、0 から 249 までの 250 種類の ID を表現できる。

ArUco マーカの一辺の長さは、訓練環境構築者がプログラムブロックを撮影する平均的な距離であると予想される 1m 前後の距離から撮影した場合に、安定して ID および姿勢を認識できる最小の大きさであった 40mm に決定した。

各パーツブロックには ArUco マーカを貼り付け、その種類を識別できるようにした。図 3.45 に、ArUco マーカの大きさおよび、ArUco マーカを貼り付けたパーツブロックの大きさを示す。パーツブロックの大きさは ArUco マーカと日本語表記の両方を貼り付けられる最小の大きさから決定した。条件ベースブロックは「<」、「>」および「=」の 3 種類があるため、条件ベースブロックに ArUco マーカを貼り付けることで、それらをシステムが識別できるようにした。図 3.46 に、ArUco マーカを貼り付けたベースブロックの大きさを示す。ベースブロックのくぼみの大きさは、パーツブロックが丁度ハマる大きさに作成し、ベースブロックの大きさは、くぼみ、ArUco マーカおよび日本語表記のすべてを含められる最小の大きさから決定した。数字パーツブロックの識別に ArUco マーカを用いると、プログラムブロック全体が大きくなってしまい、ユーザが組み立てる際に操作しにくいと感ずる可能性があるため、数字パーツブロックの識別は後述する別の方法を用いた。

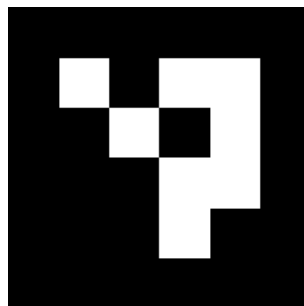


図 3.44: 本システムで使用した ArUco マーカ<sup>[33][34]</sup>の外観

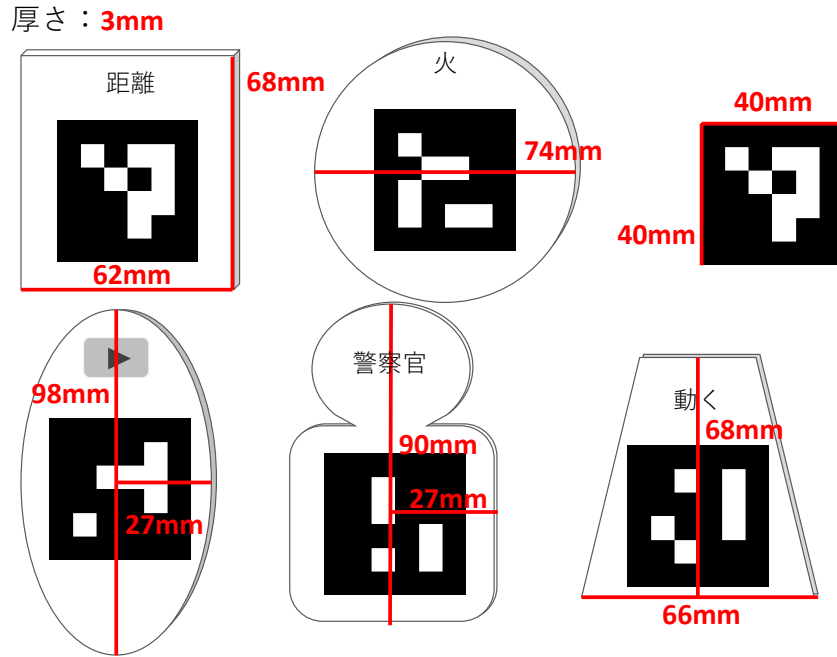


図 3.45: ArUco マーカおよびそれを貼り付けた各パーツブロックの大きさ

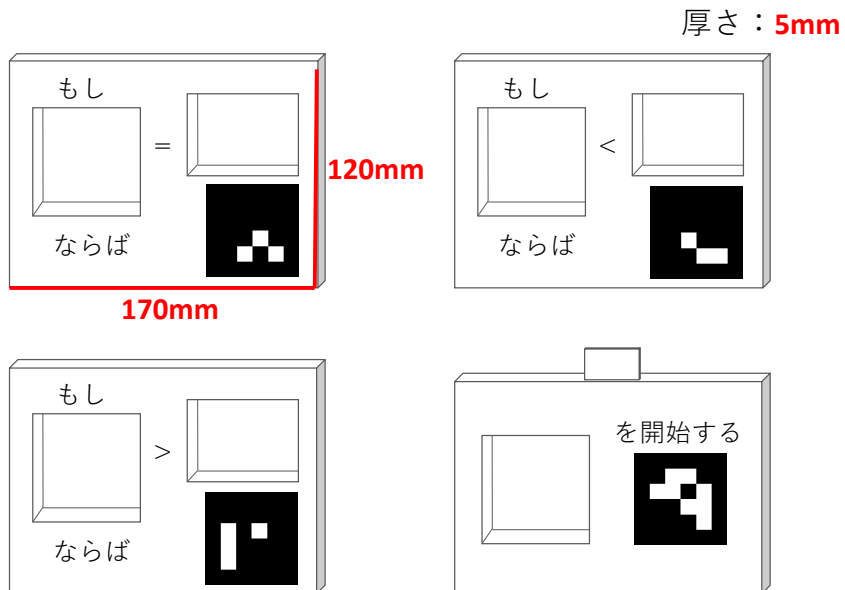


図 3.46: ArUco マーカを貼り付けたベースブロックの大きさ

前述のように、本サブシステムでは、画面内に映った複数のシナリオブロックから、1つのシナリオブロックであることを認識するために、シナリオブロック内のパーツブロックおよびベースブロックの組み合わせと位置関係を利用する。しかし、図3.19に示したように、プログラムブロックは複数の動作ブロックを下側に組み合わせられる

仕様であるため、その組み合わせの数は膨大である。しかしどのシナリオブロックにおいても、シナリオブロックの最上部の条件ブロックおよびその下に組み合わせられている動作ブロックは必ず存在するため、はじめにこれら2つの組み合わせを調べ、シナリオブロックであることを認識し、その後、下側に動作ブロックがさらに組み合わせられているかを調べることにした。以下、具体的なシナリオブロックの認識方法を述べる。

図 3.47 に示すように、シナリオブロック上には

- 条件ブロックの左側、「特定定数」の機能を示す ArUco マーカ、
- 動作ブロックの左側、「オブジェクト」の機能を示す ArUco マーカ
- 動作ブロックの右側、「動き」の機能を示す ArUco マーカ
- 条件ブロックの右側、条件ベースブロックの「演算子」の機能を示す ArUco マーカ

の4種類のマーカが存在する。そこで、表 3.9 に示すように、「特定定数」のパーツブロックには 20 未満の ID、「オブジェクト」のパーツブロックには 20 以上 40 未満の ID、「動き」のパーツブロックには 40 以上 60 未満の ID、条件ベースブロックの「演算子」には 100 以上の ID を割り当て、ID の範囲によってマーカの種類を分類することとした。以下、ID が 20 未満のマーカをマーカ 1、ID が 20 以上 40 未満のマーカをマーカ 2、ID が 40 以上 60 未満のマーカをマーカ 3、ID が 100 以上のマーカをマーカ 4 と呼ぶ。表 3.10 にマーカの ID とプログラムブロックの種類の対応を示す。

表 3.9: マーカの分類

マーカの種類名	プログラムブロックの機能	マーカの ID
マーカ 1	特定変数	0 から 19
マーカ 2	登場物体	20 から 39
マーカ 3	状態	40 から 59
マーカ 4	演算子	100 以上

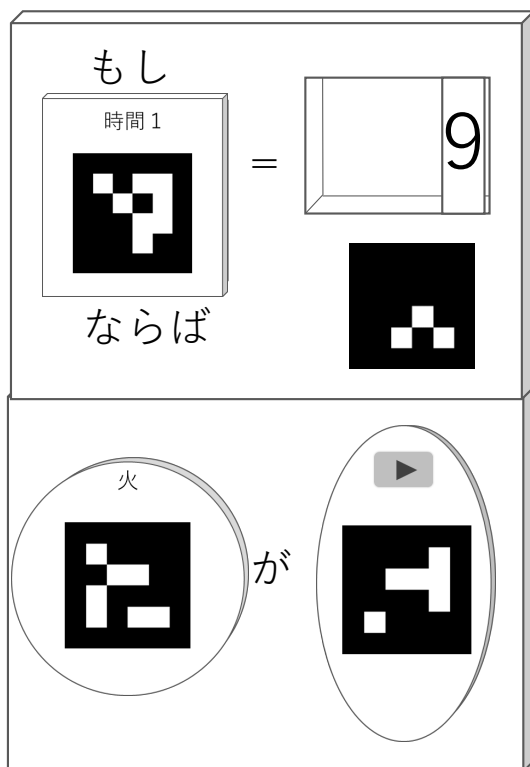


図 3.47: シナリオブロックの外観のイメージ

表 3.10: マーカの ID とプログラムブロックの種類に対応

マーカの ID	プログラムブロックの種類	マーカの ID	プログラムブロックの種類
0	距離	31	侵入者
1	時間 1	32	同僚
2	時間 2	40	再生する
3	時間 3	41	停止する
4	時間 4	50	動く
5	時間 5	51	登る
20	火	52	降りる
21	煙	53	かがむ
22	爆発	100-104	スピーカつきベースブロック「=」
23	漏水	105	スピーカつきベースブロック「<」
24	警報	106	ベースブロック「=」
25	通報	107	ベースブロック「>」
30	警察官	108	ベースブロック「<」

上述のように、1つのシナリオブロック内にはマーカ 1 からマーカ 4 が少なくとも 1

つずつ存在する。そのため、画像内にマーカ 1 からマーカ 4 が 1 つずつ存在していた場合、1 つのシナリオブロックが画像内に存在することが判別できる。しかし、図 3.48 に示すように、複数のシナリオブロックが画像内にある場合、どのマーカ 1 からマーカ 4 の組み合わせが同一のシナリオブロック上にあるかを認識する必要がある。そのため、図 3.49 に示すように、画像内の各 ArUco マーカをその ID によってマーカ 1 からマーカ 4 に分類し、別の種類のマーカとの距離をカメラで取得した depth 画像を用いて計測し、その距離が、同一シナリオブロック上のマーカ間同士の距離の最大値である 25cm 以内だった場合のみ、同一のシナリオブロック上にあると認識することとした。この処理をグルーピングと呼ぶ。

次に、シナリオブロックに 2 つ以上の動作ブロックが含まれている際の認識方法について述べる。この場合、図 3.50 に示すように、まず上部の 2 つの条件ブロックおよび動作ブロックのみグルーピングし、その後シナリオブロックの位置を認識してから下部の動作ブロックを調べる方法を採用した。動作ブロックには必ずマーカ 2 およびマーカ 3 がはめ込まれている。そこで、シナリオブロックの上部 2 つ（条件ブロックと 1 つ目の動作ブロック）の下に 2 つ目以降の動作ブロックが組み合わされているかを確認するために、後述する数字認識と同様の方法で、図 3.51 に示すように、画像から動作ブロックの範囲を切り出し、切り出した画像に対してマーカ 2 およびマーカ 3 が存在するか確認する。

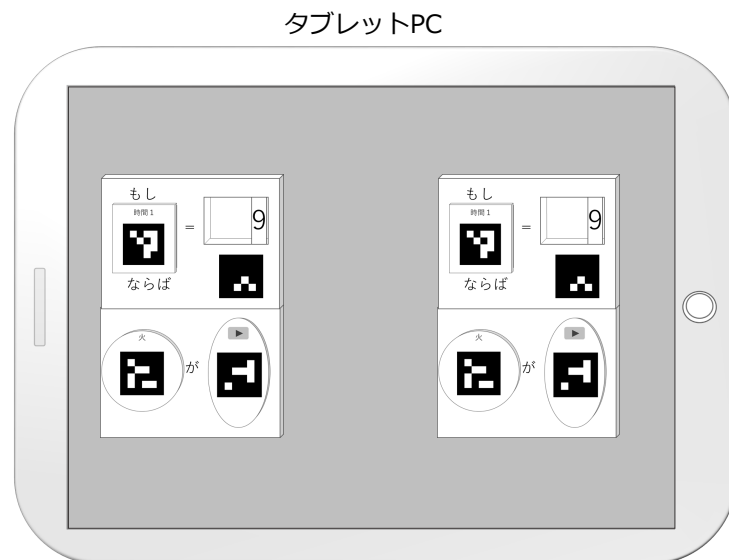


図 3.48: 複数のシナリオブロックが画像上にある様子

タブレットPC

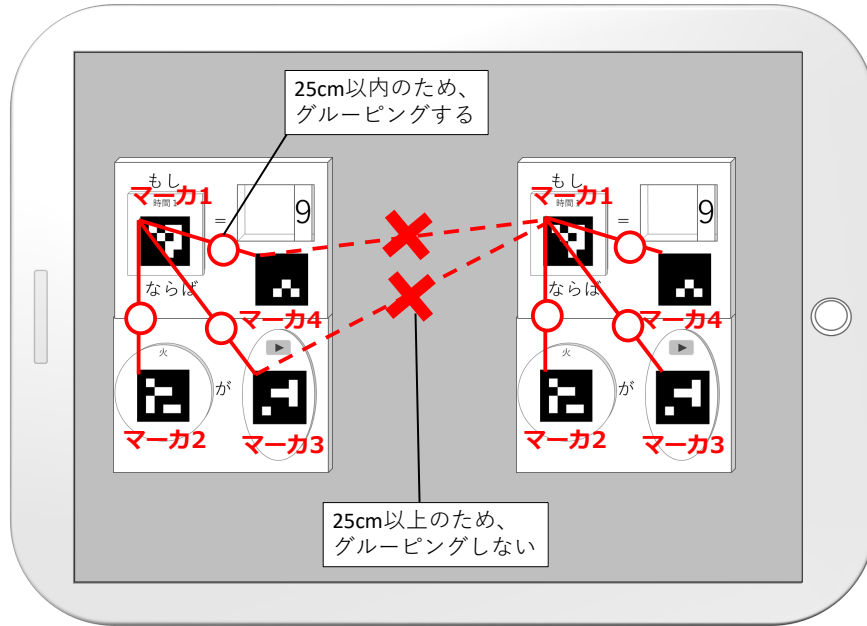


図 3.49: 各マーカの距離によってグルーピングする様子

タブレットPC

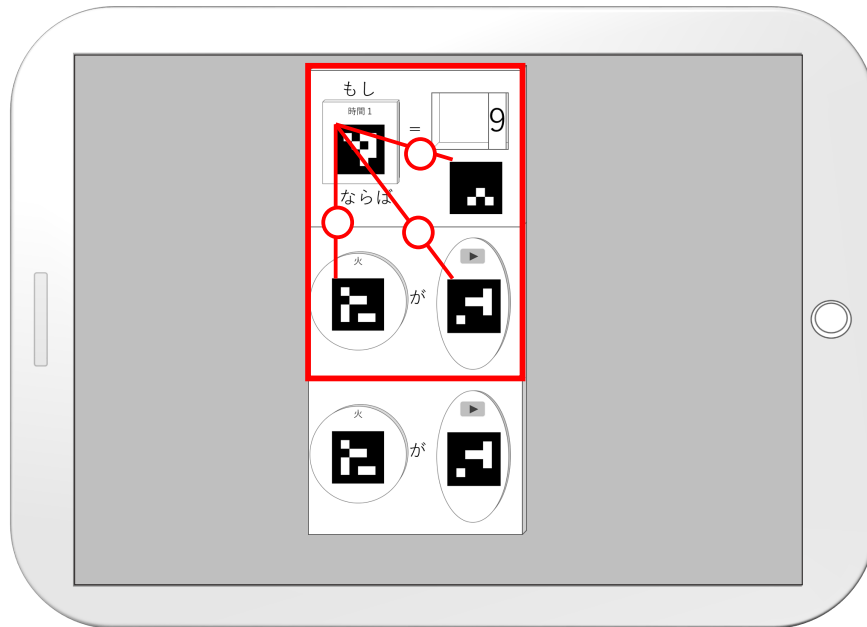


図 3.50: 上部2つのみグルーピングする様子

## タブレットPC

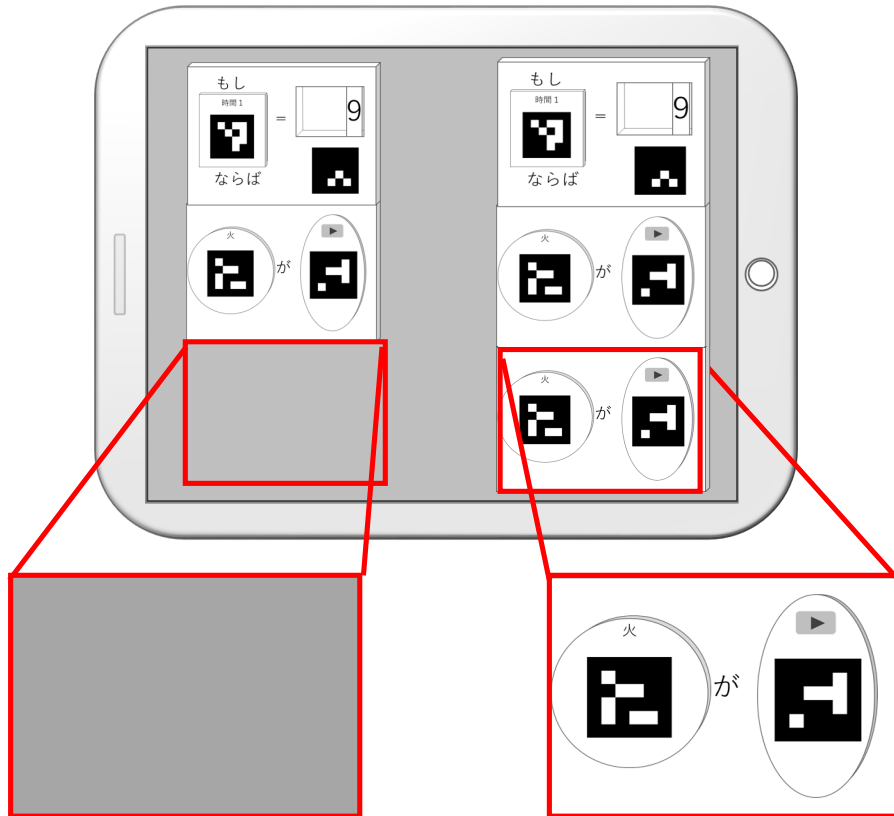


図 3.51: 動作ブロックが配置される範囲を切り出す様子

ArUco マーカを用いてシナリオブロックの姿勢を計測する方法として、本サブシステムではシナリオブロックの左上の頂点を基準とした ArUco マーカの位置情報と、ユーザが撮影した画像上に写った ArUco マーカの座標を用いて PnP 問題を解く方法を採用した。図 3.52 に PnP 問題を解くイメージを示す。まず、シナリオブロックの大きさは予め決まっているため、シナリオブロックの左上の頂点を原点  $O$  とした座標系（以下、 $O$  座標系）における、シナリオブロック上の 4 点 ( $P_1$  から  $P_4$ ) の 3 次元座標は既知である。また、ユーザが撮影した画像の左上を原点  $O'$  とした座標系（以下、 $O'$  座標系）における、シナリオブロック上の 4 点 ( $P'_1$  から  $P'_4$ ) の 2 次元座標は画像処理と座標計算により求められる。この、 $O$  座標系における  $P_1$  から  $P_4$  の 3 次元座標と、 $O'$  座標系における  $P'_1$  から  $P'_4$  の 2 次元座標を用いれば、 $O$  座標系におけるカメラの姿勢を表す同次座標行列  $R$  を計算できる。同次座標行列  $R$  の逆行列  $R^{-1}$  はカメラ姿勢を原点とした座標系（以下、カメラ座標系）におけるシナリオブロック左上の頂点の姿勢を示す。この時トラッキングによって、図 3.53 に示すようにトラッキング開始時のカメラ姿勢を原点とした座標系（以下、世界座標系）における現在のカメラ姿勢を示す同次座標

行列  $S$  を取得できる。 $S$  と  $R^{-1}$  を用いて、世界座標系におけるシナリオブロックの左上の頂点の姿勢を求められる。

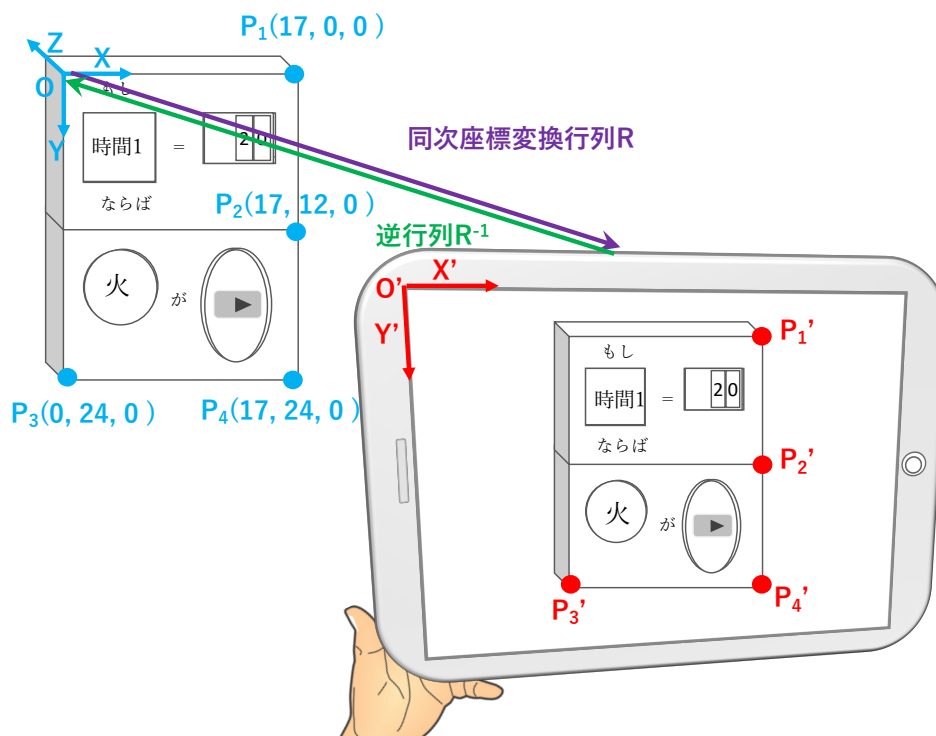


図 3.52: PnP 問題を解くイメージ

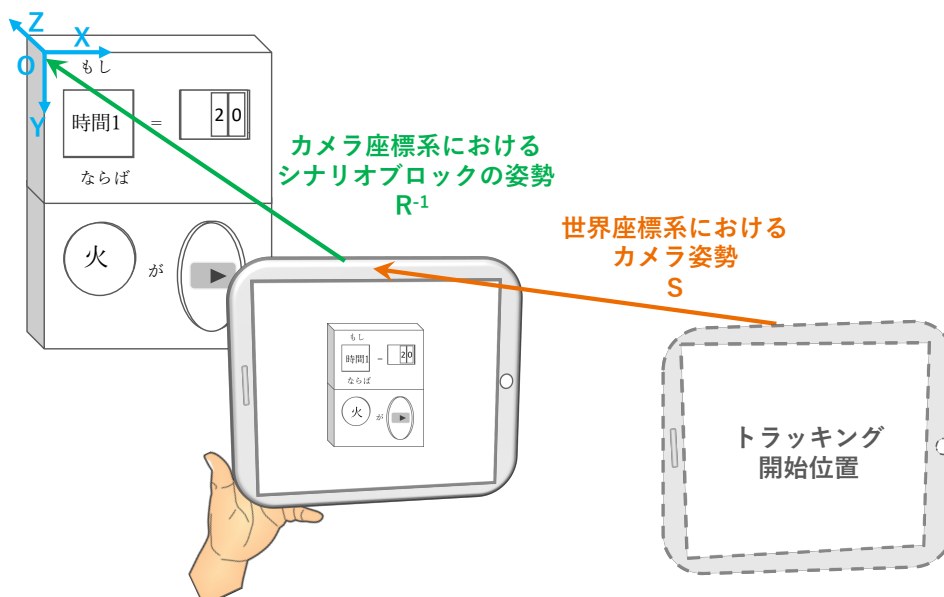


図 3.53: シナリオブロックの姿勢を計測するイメージ



以上のように、PnP問題を解くことによって、シナリオブロックの姿勢を求めることができるが、実際には、シナリオブロックの角を安定して認識することは困難である。そこで、本サブシステムでは、シナリオブロック上に配置した ArUco マーカの頂点を利用して、PnP問題を解くこととした。ArUco は、マーカの ID を認識すると同時に、その 4 頂点の  $O'$  座標系における 2 次元座標を取得できる。また、図 3.47 のように、シナリオブロック上には必ずマーカ 1 からマーカ 4 が存在し、シナリオブロック上でのマーカ 1 からマーカ 4 の位置は常に同じである。そこで、本サブシステムでは、 $O$  座標系におけるシナリオブロック上の各マーカの 4 頂点、計 16 箇所の 3 次元座標と、ユーザが撮影した画像上のシナリオブロック上の 4 つの各 ArUco マーカの 4 頂点、計 16 箇所の 2 次元座標を用いて、 $O$  座標系におけるカメラの姿勢を表す同次座標行列を求めることとした。その後、得られた同次座標行列から逆行列を算出し、トラッキング結果を用いて、世界座標系におけるシナリオブロックの姿勢を求める。ただし、本サブシステムにおいては、シナリオブロックの誤認識を防ぐために、3 フレーム連続で求めたシナリオブロックの姿勢の差が閾値以下であった場合のみ、シナリオブロックであると認識することにした。

一方、シナリオブロック内の数字パーツブロックには ArUco マーカを貼り付けていないため、図 3.54 に示すように、画像内のシナリオブロックから、数字パーツブロックが配置される領域を切り出し、切り出した画像に対して数字認識処理を行った。具体的には、まず、図 3.47 に示すように、シナリオブロックの条件ブロックの右側に位置している、数字パーツブロックの範囲の  $O$  座標系における 3 次元位置は既知である。そのため、世界座標系におけるシナリオブロックの姿勢と、トラッキング結果から得られる現在のカメラ姿勢を用いると、 $O'$  座標系における数字パーツブロックの範囲の 2 次元座標を取得できる。得られた 2 次元座標を用いて、画像を切り出し、その画像に対して数字認識処理を行った。数字認識処理には、数字認識ライブラリの tesseract ocr<sup>[37]</sup> を使用した。数字パーツブロックに使う数字のフォントには Arial を使用した。本サブシステムでは、数字認識の精度を向上させるために、3 フレーム連続で取得した数字が同じであった場合のみ、認識結果を採用することとした。

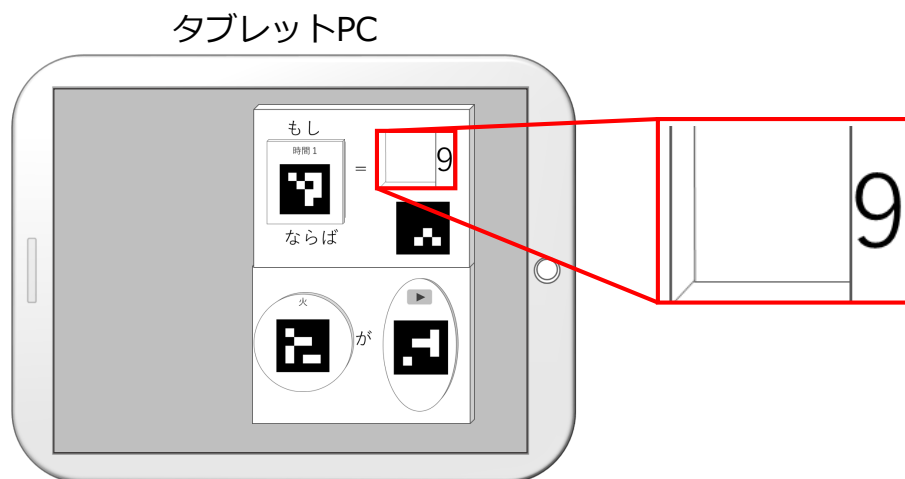


図 3.54: 数字パーツブロックが配置される領域を切り出す様子

本研究では3Dプリンタの MakerBot Replicater +を使用してプログラムブロックを作成した。表 3.11 に MakerBot Replicator +の仕様を示す。

図 3.55 に、3D プリンタで印刷したパーツブロックの外観と大きさを、図 3.56 に、3D プリンタで印刷したベースブロックの外観と大きさを示す。また、図 3.57 に ArUco マーカを貼り付けたベースブロックの例を、図 3.58 に ArUco マーカを張り付けたパーツブロックの例を示す。全パーツブロックの厚さは3mm に、全ベースブロックの厚さは5mm に設定した。数字パーツブロックの裏側には直径1cm 厚さ2mm の円形の磁石をはめ込み、ベースブロックの裏側に貼り付けた磁石と吸着するように設計した。その他の各パーツブロックには直径2mm 高さ4mm の円柱状の突起を付け、ベースブロックのくぼみに空けた穴に差し込むことで、各パーツブロックと各ベースブロックを組み合わせた際にパーツブロックがずれないようにした。

表 3.11: MakerBot Replicator +の仕様

造形方式	熱溶解積層法方式
積層ピッチ	0.1mm (100 $\mu$ m)
モデル材料	MakerBot PLA フィラメント
造形サイズ	(X)295mm × (Y)195mm × (Z)165mm
制御ソフトウェア	MakerBot Print

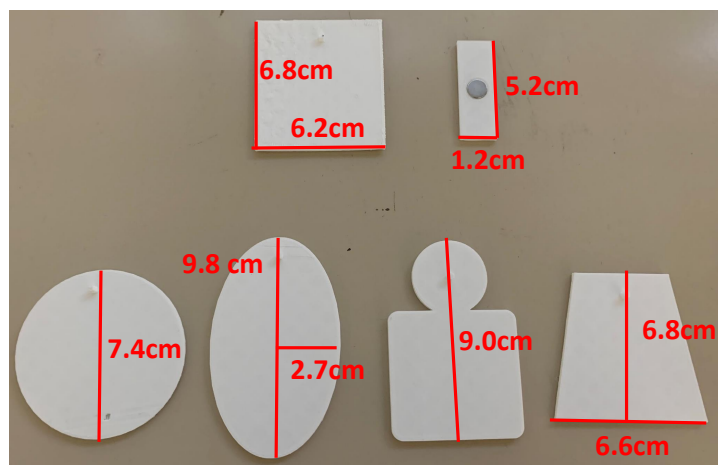


図 3.55: 3D プリンタで印刷したパーツブロックの外観とその大きさ

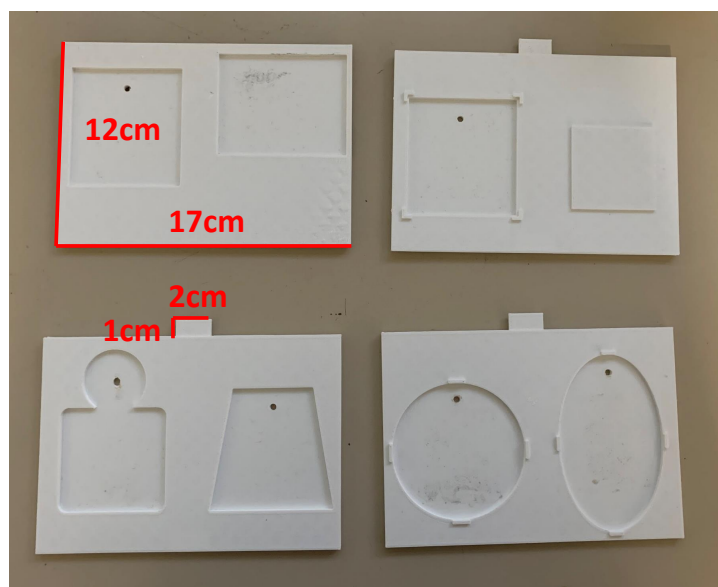


図 3.56: 3D プリンタで印刷したベースブロックの外観とその大きさ

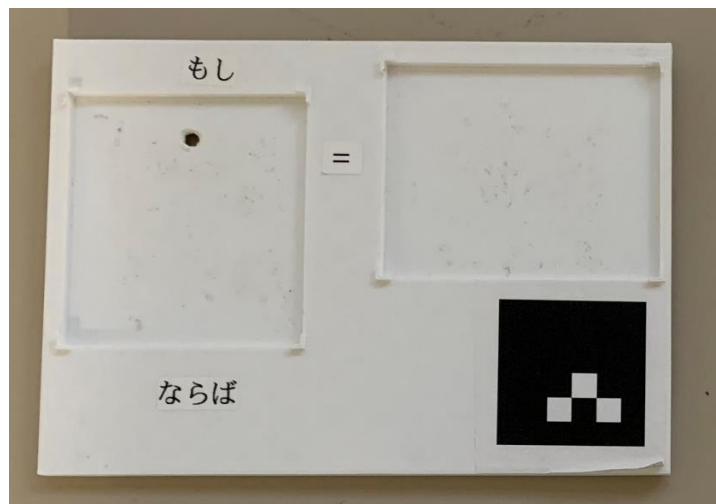


図 3.57: ArUco マーカを貼り付けたベースブロックの例

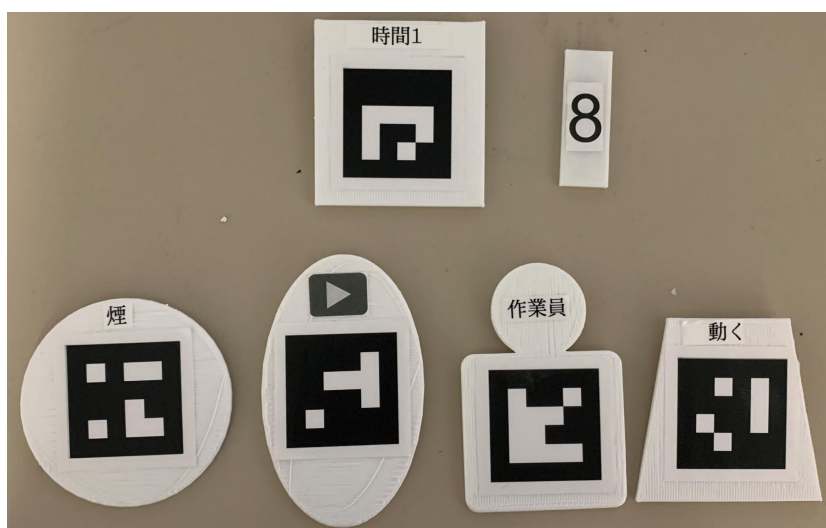


図 3.58: ArUco マーカを貼り付けたパーツブロックの例

本サブシステムでは、認識したシナリオブロックの姿勢および内容をシナリオファイルとしてテキストファイルに出力し、訓練環境体験サブシステムで読み込めるようにした。シナリオファイルのフォーマットとその例を付録 C に示す。

### 3.6.2 訓練環境体験サブシステムの実装

本項では、3.5 節で述べた設計に基づく訓練環境体験サブシステムの実装について述べる。図 3.59 に、訓練環境体験サブシステムのハードウェア構成を示す。本サブシス

テムは、高性能ノート PC、タブレット PC、RGB-D カメラ、ルータ、小型 Bluetooth スピーカおよび Raspberry Pi で構成する。高性能ノート PC、タブレット PC、RGB-D カメラおよびルータは訓練環境構築サブシステムと同じものを用いた。3.5 節で述べたように、本システムでは、効果音を体験者が把持するデバイスからではなく、分散して配置されるプログラムブロックから流すため、準備の利便性を考慮すると、無線接続可能なスピーカを選定する必要がある。無線 LAN 接続のスピーカは大きさが大きく、プログラムブロックに取り付けることが困難だったため、プログラムブロックに取り付けられる大きさの小型の Bluetooth スピーカである EWA A109mini を用いた。ただし、一般的な PC が Bluetooth 接続できるスピーカは 1 台のみであるため、高性能ノート PC と複数のマイクロコンピュータの Raspberry Pi をハブを介して有線接続し、各 Raspberry Pi と 1 台のスピーカを Bluetooth 接続した。

表 3.12、表 3.13、図 3.60 に本システムで用いた Raspberry Pi の仕様、本システムで用いた Bluetooth スピーカの仕様、Bluetooth スピーカの外観を示す。今回の実装では、Raspberry Pi および Bluetooth スピーカを 5 台ずつ用いたが、原理的には使用可能な IP アドレスの個数だけ同時接続することが可能であり、構築するシナリオの規模に応じて台数を変更可能である。

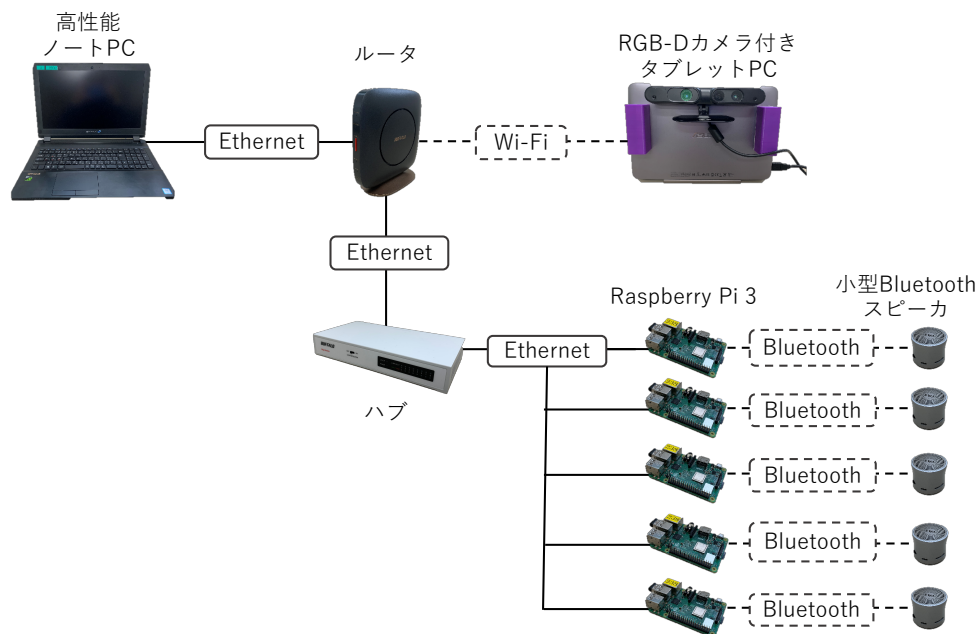


図 3.59: 訓練環境体験サブシステムのハードウェア構成

表 3.12: 本システムで用いた Raspberry Pi の仕様

製品名	Raspberry Pi 3 Model B+ (UD-RP3BP)
CPU	Broadcom BCM2837B0, Cortex-A53 (ARMv8) SoC
メモリ	1GB
大きさ	約 86 (W) × 57 (D) × 17 (H) mm
重さ	約 50g



図 3.60: 本システムで用いた Bluetooth スピーカの外觀

表 3.13: 本システムで用いた Bluetooth スピーカの仕様

製品名	EWA A109mini
動作範囲	10m
連続再生時間	約 2 時間
大きさ	DIA.48 × H38.5mm
重さ	約 210g
Bluetooth 規格	V3.0

図 3.61 に、タブレット PC、高性能のノート PC、Raspberry Pi 上で実行するプログラムとデータの流れを示す。タブレット PC 上では、訓練環境構築サブシステムと同様に、color 画像と depth 画像の取得および、高性能ノート PC と通信を行うクライアントアプリを実行する。高性能ノート PC では、他のアプリケーションとの通信、トラッキングおよび訓練シナリオの管理を行う体験サーバアプリと、サーバアプリから画像や命令を受け取り、CG の描画および合成画像のキャプチャを行うレンダリングアプリを実行する。Raspberry Pi では、高性能ノート PC と通信および、音声の再生を行うサウンドアプリを実行する。

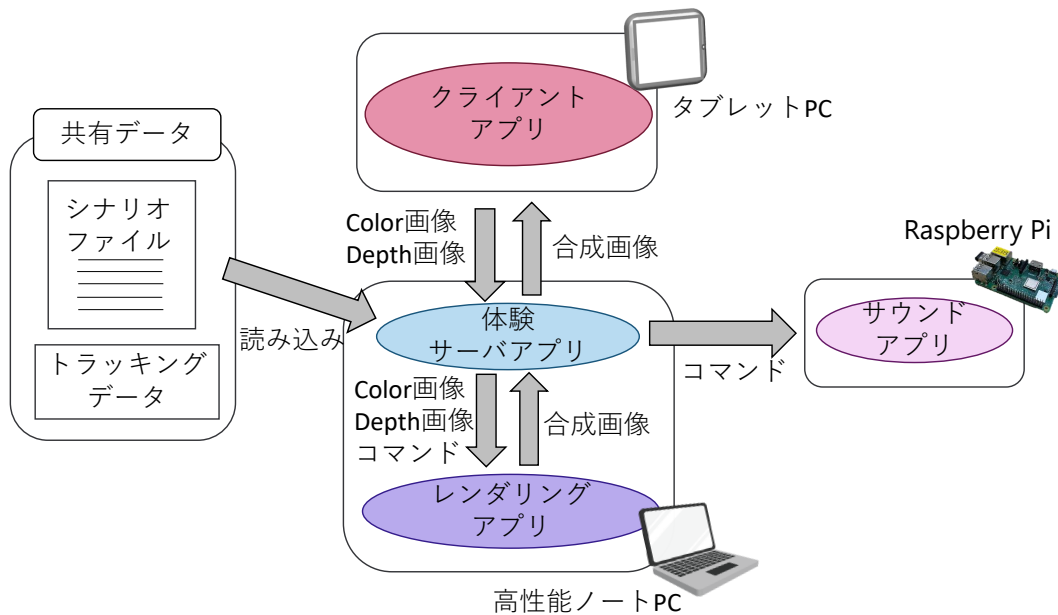


図 3.61: 訓練環境体験サブシステムで実行するプログラムとデータの流れ

体験サーバアプリはクライアントアプリから RGBD 画像を受け取りトラッキング処理を行うとともに、訓練シナリオの管理を行う。体験サーバアプリは Microsoft Visual Studio 2017 を用いて開発し、開発言語は C++ を用いた。サーバアプリと他すべてのアプリケーションの通信には、訓練環境構築サブシステムと同じく boost1.69.0-1 の asio を用いた。

レンダリングアプリでは、サーバアプリから RGBD 画像や描画命令を受け取り、CG を描画したのち、描画結果をサーバアプリへ送る。レンダリングアプリは Unity 2017 と Microsoft Visual Studio 2017 を用いて開発し、開発言語は C# を用いた。以下、レンダリングアプリでの処理の詳細を述べる。RGB-D カメラで撮影した color 画像上に CG を適切に重畳表示した合成画像を作成するためには、画像を取得した時の RGB-D カメラの姿勢に応じて適切な方向から見た CG をレンダリングし、その結果を実画像上の適切な位置に表示する必要がある。そのためには仮想空間の座標とトラッキングの座標が統一されており、RGB-D カメラの姿勢とレンダリング時の仮想カメラの姿勢が同じである必要がある。

仮想空間の座標とトラッキングの座標を統一するために、本研究では、訓練環境構築サブシステムにおける撮影開始時のカメラ姿勢を世界座標系の原点として、各アプリケーションで統一した。特に、Unity では、実空間と同じスケールの座標系をもつ仮想空間を作成し、この仮想空間内に、シナリオブロックの姿勢に応じて 3D モデルを配

置した。また、図 3.62 に示すように、仮想空間内に置いた仮想カメラの姿勢を、実空間における RGB-D カメラの姿勢と同じにした上でレンダリングを実行し、その結果と RGB-D カメラで撮影した color 画像を組み合わせることによって、合成画像を作成する。

この方法によって CG の位置合わせは可能であるが、体験者からみて 3D モデルと実世界の物体のどちらが手前にあるかを考慮し、奥にある物体が手前にある物体の影に隠れて見えない状態（以下、オクルージョン）を考慮していない。図 3.63 の左図に示すように、オクルージョンを考慮していない画像では、本来は、3D モデルの手前に物体があり、3D モデルが見えないはずの場合でも、3D モデルが体験者から見えてしまう。このような画像では、3D モデルが正しく位置合わせされていても、距離感を掴みづらく、3D モデルが実空間のどこに位置しているのかがわかりづらくなる。一方、図 3.63 の右図に示すように、オクルージョンを考慮して映像を合成した場合、実空間で 3D モデルの手前に視界を遮る物体があった場合、3D モデルは物体に隠れて見えなくなる。このようなオクルージョンを考慮している合成画像の方が現実感が高いと考えられる。そこで本サブシステムでは、図 3.64 に示すように、CG のレンダリング時に生成される仮想カメラ画像における各ピクセルの深度情報と、実空間の depth 画像における各ピクセルの深度情報を比較することによって、仮想カメラ画像と color 画像のピクセルのうち、手前にある方のみ描画し、オクルージョンを考慮した合成画像を作成するようにした。



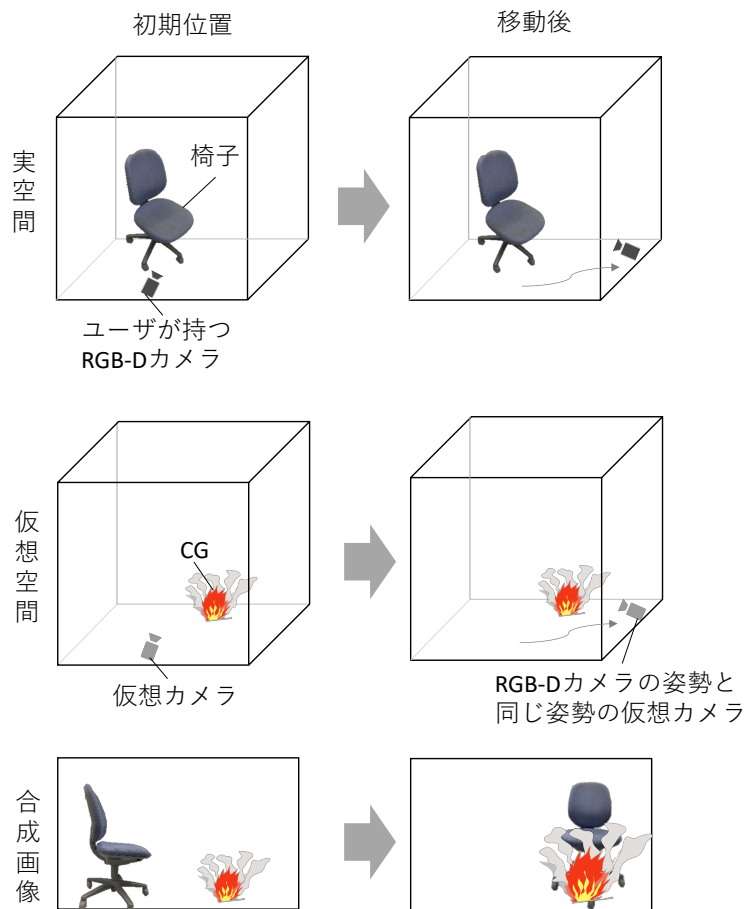


図 3.62: 実空間のカメラと同時に仮想カメラが移動し合成画像を作成する様子

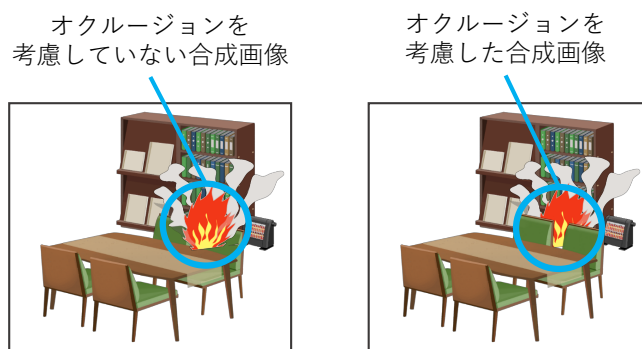


図 3.63: オクルージョンを考慮していない画像としている画像のイメージ

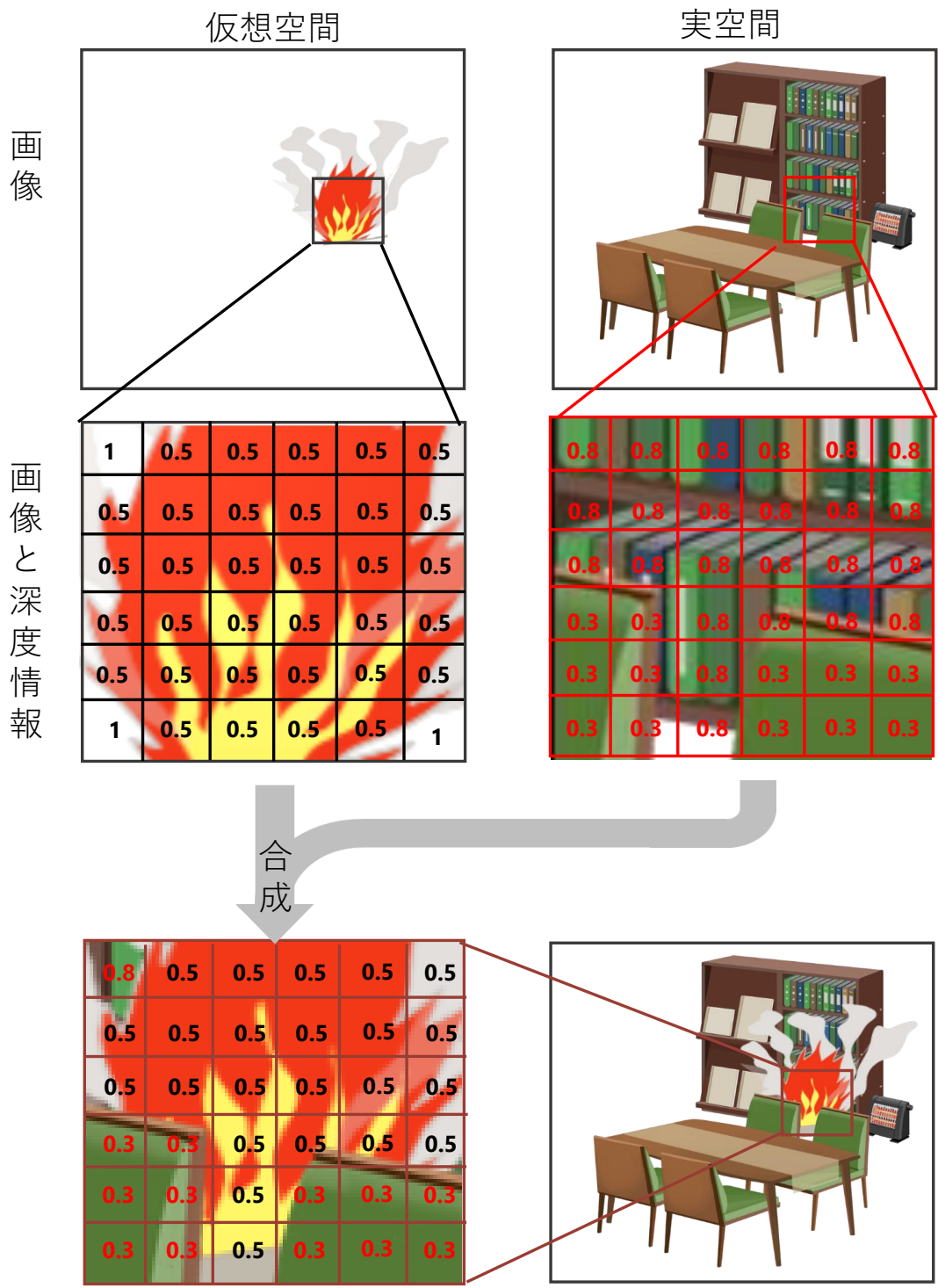


図 3.64: Zバッファを用いて合成画像を作成する様子

図 3.65 にレンダリングアプリの処理の流れを示す。はじめに、レンダリングアプリでは、体験サーバアプリから送信されたコマンドに従い、3D モデルを仮想空間内に配置する。メインループでは、はじめに体験サーバアプリからコマンドが送信されている

かどうかを確認し、コマンドを受信した場合、コマンドに従い、CGの再生、停止、移動、表示、および非表示を行う。その後、color画像、depth画像および現在のRGB-Dカメラの姿勢を体験サーバアプリから受信し、それらを元にcolor画像上にCGを重畳表示したAR表示用の画像を合成する。その後、合成画像を体験サーバアプリに送信する。

炎や水、警察官などの3Dモデルは、Unity Asset Store<sup>[35]</sup>から入手可能なものを用いた。また、人CGのアニメーション作成には、mixamo<sup>[36]</sup>を用いた。

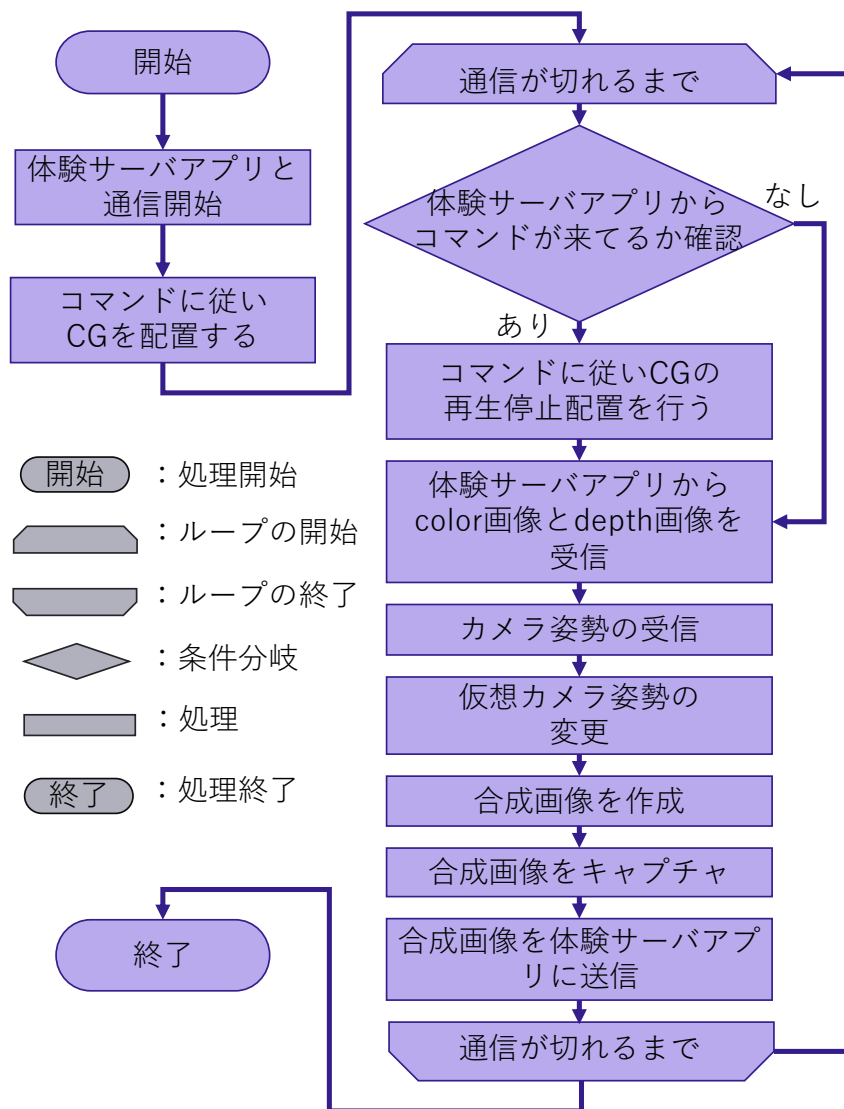


図 3.65: レンダリングアプリの処理の流れ

Raspberry Pi 上で実行するサウンドアプリは、体験サーバアプリからコマンドを受信し、Bluetooth 接続したスピーカから効果音を流す。サウンドアプリの開発にはC++

を用いた。図 3.66 にサウンドアプリの処理の流れを示す。Bluetooth 接続のスピーカは、バッテリーの持続時間を長くするため、一定時間、音が流れないと自動的にスリープモードに入るものが多い。今回も、スピーカが訓練体験中にスリープモードに入らないようにするために、効果音を流していない間も無音の音声を流し続ける仕様とした。

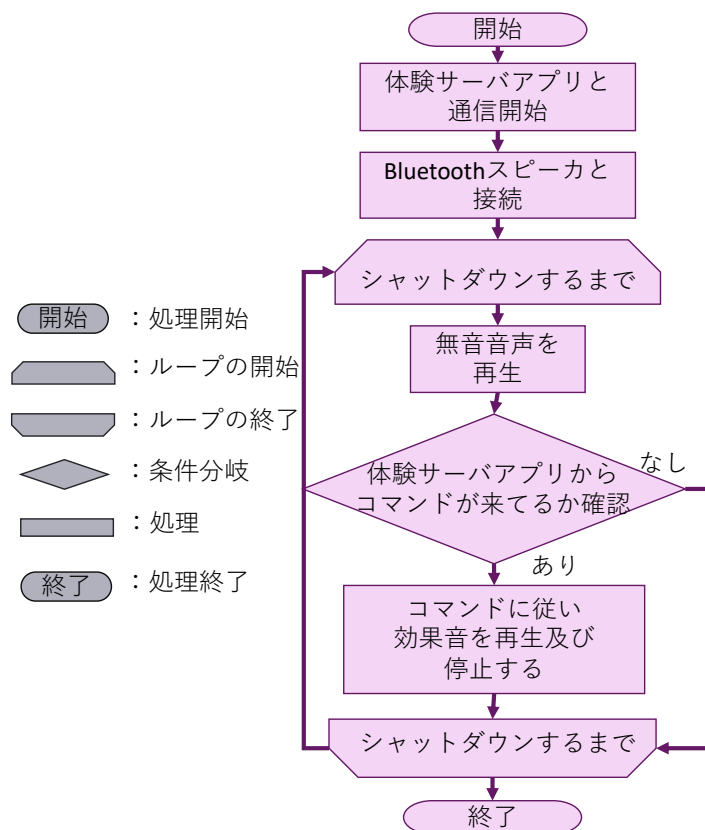


図 3.66: サウンドアプリの処理の流れ

### 3.6.3 ARTMの動作例

図 3.67 に訓練環境構築サブシステムを用いてシナリオブロックを撮影した際に表示される画面の例を示す。図 3.67 に示すように、シナリオブロックが正しく認識されると、そのブロックの周囲に緑色の線が描画され、正しく認識したことをユーザに知らせている。

また、訓練環境体験サブシステム使用時の画面の例を図 3.68、図 3.69 に示す。図 3.68 に示すように、シナリオブロックを設置した場所に火が AR 表示され、火災を模擬できる。また、図 3.69 に示すように、画像の奥と手前に配置されたシナリオブロックに従い、警察官が移動する様子を模擬できる。

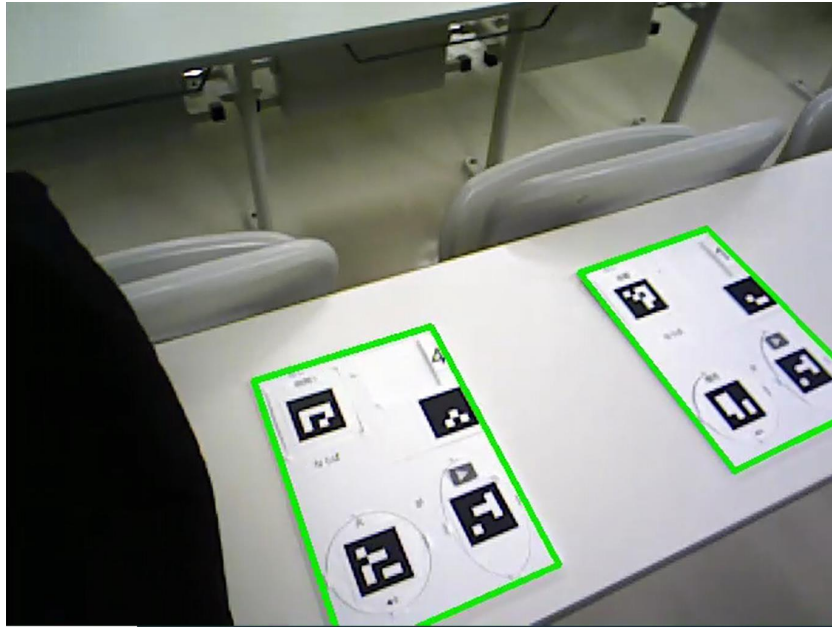


図 3.67: シナリオブロック撮影時の画面の例



図 3.68: 火の CG を重畳表示した画面



図 3.69: 移動する人の CG を重畳表示した画面

## 第 4 章 システムの評価

本章では、アンケートおよびインタビューにより行った、ARTM 使用時の理解および操作の容易さと訓練シナリオの自由度に関する評価について述べる。まず評価の目的と概要について述べ、次に評価の方法について述べる。その後、評価の結果について述べ、最後に結果の考察について述べる。

### 4.1 評価の目的と概要

第 3 章で述べたように、本研究で開発した ARTM の目標は、以下の 3 つである。

- プログラミングの高度なスキルを持っていない人でも容易に理解および操作ができること
- 訓練を実施する現場を直接観察しながら訓練環境を構築できること
- 現場で求められている水準の複雑さの訓練シナリオを作成できること

この内、訓練を実施する現場を直接観察しながら訓練環境を構築することは、プログラムブロックによって可能になった。しかし、他 2 つの目標に関しては、達成できているかを評価する必要がある。

そこで、本評価は、プログラミングの高度なスキルを持っていない人でも ARTM を容易に理解および操作ができること、ARTM で現場で求められている水準の複雑さの訓練シナリオを作成できることの 2 つを評価することを目的とする。以下では、プログラミングの高度なスキルを持っていない人でも容易に ARTM の理解および操作ができることを ARTM の容易さ、現場で求められている水準の訓練シナリオを作成できることを訓練シナリオの自由度と呼ぶ。

本評価では、ARTM の使用想定者である重要施設の作業員として、京都大学複合原子力科学研究所の職員に、ARTM を用いて AR 訓練環境の構築および体験をしてもらった。ARTM を使用后、評価者にアンケートおよびインタビューに回答してもらうことにより、ARTM の容易さと、訓練シナリオの自由度を評価した。

本評価では、ARTM の容易さと、訓練シナリオの自由度を、ヒューリスティック法<sup>[38]</sup>を用いて評価した。ヒューリスティック法とは、システムのプロトタイプやその仕様

書を、評価者自身の直感とインタフェース設計に関する常識によって評価し、インタフェースの問題点を探し出す方法である。評価者の人数は数人程度が適当とされる<sup>[38]</sup>。ヒューリスティック法による評価の利点は、システムの問題点を効率的に収集でき、その際のコストを低く抑えられることである。

## 4.2 評価方法

### 4.2.1 評価項目

ARTMの容易さを評価するために、ユーザがARTMの使用方法を容易に理解できることおよび、ユーザがARTMを思った通りに操作できることを評価する必要がある。以下では、ユーザが使用方法を容易に理解できることを認知性、ユーザが思った通りに操作できることを操作性と呼ぶ。

しかし、アンケートおよびインタビューでARTMのすべての機能の認知性および操作性を評価することは難しい。そこで、ARTMにおいて重要度が高い以下の項目に絞って評価することとした。

1. プログラムブロックの機能に関する認知性

プログラムブロックを用いて何を作成できるかを容易に理解できる。

2. プログラムブロックの組み立てに関する認知性

どのようにしたらプログラムブロックを組み立てられるかを容易に理解できる。

3. プログラムブロックの組み立てに関する操作性

プログラムブロックを思った通りに組み立てられる。

4. プログラムブロックの設置に関する認知性

どのようにしてプログラムブロックを設置したらよいか容易に理解できる。

5. プログラムブロックの設置に関する操作性

プログラムブロックを思った通りに設置できる。

6. プログラムブロックの撮影に関する操作性

プログラムブロックを思った通りに撮影できる。



## 7. ARTM 全体の認知性および操作性

ARTM をどのように使用するのかが容易に理解でき、思った通りに操作できる。

また、訓練シナリオの自由度を評価するために、現場で求められている水準の複雑さの訓練シナリオを作成するために必要な機能が ARTM に備わっているかどうかを評価する必要がある。そこで、ARTM の機能における以下の項目が十分かどうかを評価することとした。

### 1. CG の種類

3.3.2 項で述べたプログラムブロックにおける「オブジェクト」の種類。

### 2. CG の動きの種類

3.3.2 項で述べたプログラムブロックの機能における「動き」、「見た目」および「音」の種類。

### 3. 条件の種類

3.3.2 項で述べたプログラムブロックの機能における「特定変数」、「演算子」および「制御」の種類。

## 4.2.2 課題内容

本項では、ARTM の容易さを評価するために策定した課題であるシナリオ作成課題および、訓練シナリオの自由度を評価するために策定した課題である自由作成課題の2つの課題について述べる。

4.2.1 項で述べた ARTM の容易さに関する7つの項目を評価するために、7つの項目に関わる ARTM の機能の評価者が全て使用するよう、実験者が予めシナリオ作成課題を策定した。ただし、1つの訓練シナリオに全ての項目に関わる機能を取り入れると、訓練シナリオが複雑になってしまうため、訓練シナリオを2つに分け、各訓練シナリオを評価者に構築してもらった。以下、作成した2つの課題をシナリオ作成課題1とシナリオ作成課題2と呼ぶ。図 D.1 に評価者が参照したシナリオ作成課題1の課題内容を記した資料を、図 D.2 にシナリオ作成課題2の課題内容を記した資料を示す。

シナリオ作成課題1は、屋内での漏水を模擬する課題とした。具体的には、「訓練開始からの経過時間が6秒後、20秒後になれば2か所で漏水のCGが再生され、訓練開始からの経過時間が25秒、30秒になればそれらの漏水のCGが停止される」という、

「特定変数」内の「時間1」、「オブジェクト」内の「漏水」、「動き」内の「再生される」および「停止される」を使用する課題内容とした。プログラムブロックではCGの位置だけでなく、向きも設定する必要があるため、シナリオブロックの向きに対してどのような向きでCGが表示されるかを、シナリオ作成課題1の課題内容を記した資料に明記した。

シナリオ作成課題2は、「部屋に侵入者が侵入し、部屋内の紙類に放火する。それが燃え広がり爆発性物体が爆発する。訓練体験者が通報することで警察が来る」という状況を模擬する課題とした。具体的には侵入者が侵入する様子、侵入者がかがんで火をつける様子、侵入者が去る様子、火が発生する様子、火が燃え広がる様子、爆発が発生する様子、ユーザが通報する様子、通報の1分後に警察が駆けつける様子を設定する課題内容となる。火が発生する様子、爆発が発生する様子およびユーザが通報する様子に関しては、シナリオブロックの設置位置を、シナリオ作成課題2の課題内容を記した資料内で指定した。「特定変数」内の「時間1」、「時間2」および「距離」、「オブジェクト」内の「火」、「爆発」、「通報」、「侵入者」および「警察官」、「動き」内の「再生される」および「動く」および「かがむ」を使用する課題内容となる。

自由作成課題は、4.2.1項で述べた訓練シナリオの自由度に関する3つの項目を評価するために、評価者が自ら考案した訓練シナリオをARTMを用いて作成できるか試みてもらった課題とした。図D.3に自由作成課題を説明した資料を示す。

評価の際には、はじめに、評価者が求めるAR訓練を考案してもらった。その後、考案してもらった訓練シナリオをARTMを用いて作成できるか試してもらったことにより、訓練シナリオの自由度に関する3つの項目が十分かどうかを評価した。

### 4.2.3 評価環境

本評価は、京都大学複合原子力科学研究所の会議室で実施した。図4.1に評価に使用した部屋の様子、図4.2に、評価に使用した部屋の見取り図を示す。シナリオ作成課題2において、訓練環境を構築する際にシナリオブロックを設置する位置の目印として使用する電話、紙類および爆発性物体を評価部屋内に設置した。

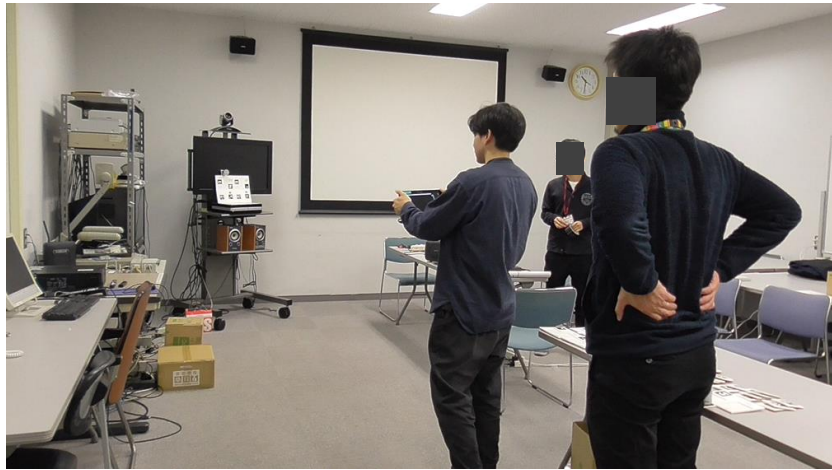


図 4.1: 評価に使用した部屋の様子

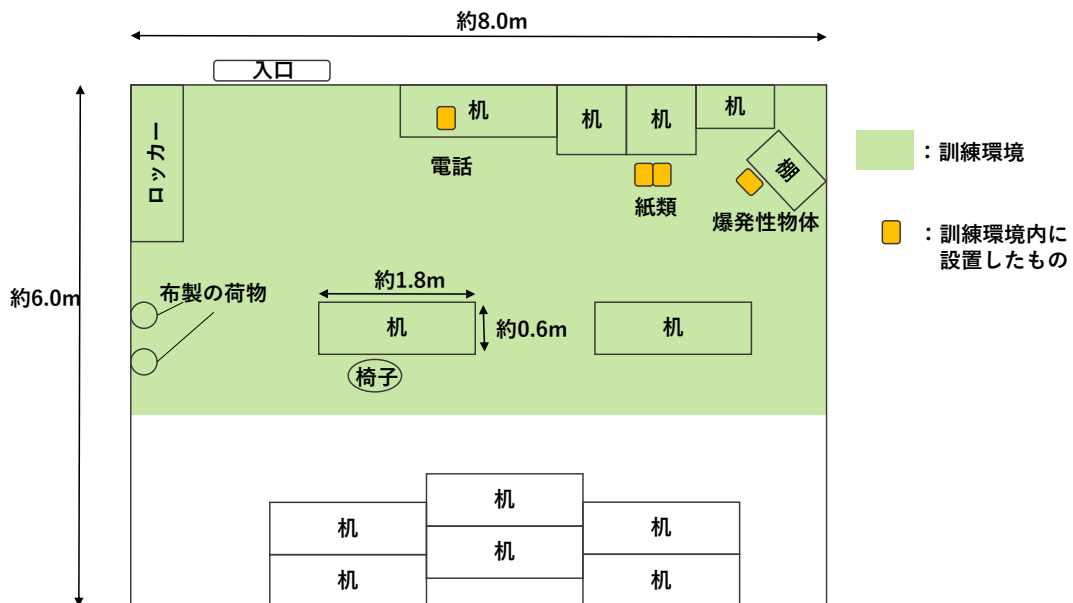


図 4.2: 評価に使用した部屋の見取り図

#### 4.2.4 評価手順

評価の際には、評価者の時間の都合上、以下に述べる手順1もしくは手順2の2つの手順を実施した。

図 4.3 に本評価の手順1を示す。手順1では、評価者は全ての手順を1人で行った。手順1では、はじめに本評価の目的および流れを、付録図 D.4 に示した評価概要の事前説明資料を用いて、約3分で実験者が口頭で評価者に説明した。次に ARTM の使用

方法を評価者に理解してもらうために、図 D.5 に示した、ARTM のシステム構成および各サブシステムの使用フローを説明した約 2 分の説明ビデオを評価者に視聴してもらった。説明ビデオ内容の補足資料を図 D.6 に示す。その後、図 D.7 から図 D.11 および図 A.1 から図 A.7 に示した、プログラムブロックの概要およびプログラムブロックを用いて作成できるシナリオを記した説明資料を約 15 分で評価者に読んでもらった。この際、質問がある場合は実験者が口頭で回答した。その後、ARTM の使用練習として、プログラムブロックの説明資料に示した単純な訓練環境を構築および体験してもらった。

練習後、約 15 分間シナリオ作成課題 1 に取り組んでももらった。プログラムブロックの組み立てから、AR 訓練の体験まで終了した時点で課題終了とした。その後、約 30 分間シナリオ作成課題 2 に取り組んでももらった。シナリオ作成課題 1 と同様に、プログラムブロックの組み立てから、AR 訓練の体験まで終了した時点で課題終了とした。シナリオ作成課題終了後約 5 分で、実験者が ARTM の容易さに関するアンケートを評価者に対して実施し、各アンケート項目の回答理由について実験者が評価者にインタビューした。

その後、約 30 分自由作成課題に取り組んでももらった。自由作成課題終了後約 5 分で、実験者が訓練シナリオの自由度に関するアンケートを評価者に対して実施し、各アンケート項目の回答理由について評価者にインタビューした。



図 4.3: 本評価の手順 1

図 4.4 に本評価の手順 2 を示す。手順 2 は、複数の評価者で同時に実施した。手順 2 では、手順 1 と同様に、本評価の目的および流れを、事前説明資料を用いて、約 3 分で実験者が口頭で評価者に説明し、その後、ARTM の使用方法を評価者に理解してもらうために、ARTM のシステム構成および各サブシステムの使用フローを説明した約 2 分の説明ビデオを実験者が評価者に見せた。次に、手順 1 と同様に、プログラムブロックの説明資料を約 15 分で評価者に読んでもらった。この際、質問がある場合は実験者が口頭で回答した。その後、手順 1 と同様に、ARTM の使用練習として、プログラムブロックの説明資料に示した単純な訓練環境を構築および体験してもらった。練習後、約 15 分のシナリオ作成課題 1 もしくは約 30 分のシナリオ作成課題 2 のどちらかに取り組んでもらった。シナリオ作成課題 1 に取り組む手順を手順 2-1、シナリオ作成課題 2 に取り組む手順を手順 2-2 と呼ぶ。各課題終了後約 5 分で、実験者が ARTM の容易さに関するアンケートを実施した。一部の評価者に対しては各アンケート項目の回答理由についてインタビューした。

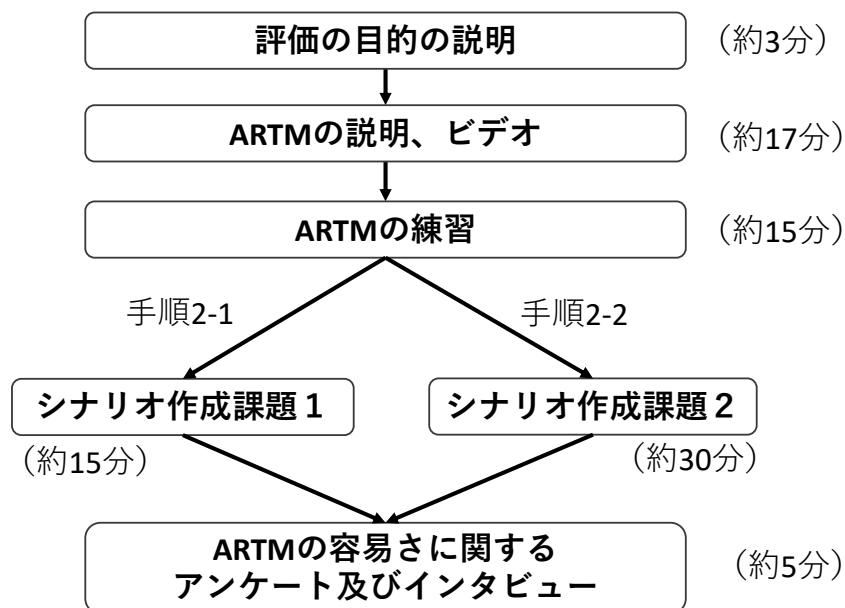


図 4.4: 本評価の手順 2

#### 4.2.5 評価アンケート

本項では、ARTMの容易さおよび訓練シナリオの自由度を評価するために実施したアンケートについて述べる。本評価では、シナリオ作成課題の後にARTMの容易さに関するアンケートを、自由作成課題の後に訓練シナリオの自由度に関するアンケートを実施した。評価者には各アンケート項目に対して「そう思わない」、「あまりそう思わない」、「どちらでもない」、「ややそう思う」、「そう思う」の5段階で回答してもらった。ARTMの容易さおよび訓練シナリオの自由度に関するアンケート用紙を図D.12および図D.13にそれぞれ示す。また、容易さに関するアンケート項目とARTMの認知性および操作性の対応と、訓練シナリオの自由度に関するアンケート項目とARTMの機能性の対応を、表4.1および表4.2にそれぞれ示す。

表 4.1: 容易さに関するアンケート項目と ARTM の容易さに関する項目の対応

番号	アンケート項目	システムの認知性および操作性
(X)-1	どのパーツブロックがどの機能を表すか理解できた。	プログラムブロックの機能に関する認知性
(X)-2	ベースブロックにどのパーツブロックがはめ込められるか理解できた。	プログラムブロックの組み立てに関する認知性
(X)-3	時間 1 を条件として設定することは容易だった。	プログラムブロックの組み立てに関する認知性および操作性
(X)-4	距離を条件として設定することは容易だった。	プログラムブロックの組み立てに関する認知性および操作性
(X)-5	時間 1 と時間 2 の違いを理解することは容易だった。	プログラムブロックの機能に関する認知性
(X)-6	物 CG (火や漏水など) の再生を設定することは容易だった。	プログラムブロックの組み立てに関する認知性および操作性
(X)-7	物 CG (火や漏水など) の停止を設定することは容易だった。	プログラムブロックの組み立てに関する認知性および操作性
(X)-8	人物 CG (警察官、侵入者など) の動きを設定することは容易だった。	プログラムブロックの組み立てに関する認知性および操作性
(X)-9	CG の表示位置を設定することは容易だった。	プログラムブロックの設置に関する認知性および操作性
(X)-10	CG の表示向きを設定することは容易だった。 (漏水の流れや、人の立つ向きなど)	プログラムブロックの設置に関する認知性および操作性
(X)-11	シナリオブロックを撮影することは容易だった。	プログラムブロックの入力方法に関する操作性
(X)-12	想像していた通りに CG が表示された。	ARTM 全体の認知性および操作性
(X)-13	初めて使う人でも簡単にシステムを使用できる。	ARTM 全体の認知性および操作性

表 4.2: 訓練シナリオの自由度に関するアンケート項目と訓練シナリオの自由度に関する項目の対応

番号	アンケート項目	プログラムブロックの機能
(Y)-1	現場で必要とされている訓練を作成するのに必要な物 CG (火、漏水など) を備えている。	CG の種類
(Y)-2	現場で必要とされている訓練を作成するのに必要な人 CG (警察官など) を備えている。	CG の種類
(Y)-3	現場で必要とされている訓練を作成するのに必要な物 CG の動き (再生、停止) を備えている。	CG の動きの種類
(Y)-4	現場で必要とされている訓練を作成するのに必要な人 CG の動き (動く、乗り越える、飛び降りる、かがむ) を備えている。	CG の動きの種類
(Y)-5	現場で必要とされている訓練を作成するのに必要な条件 (時間、ユーザーとの距離など) を備えている。	条件の種類

## 4.2.6 評価者

本評価の評価者は、京都大学複合原子力発電所の職員7名(a、b、c、d、e、f、g)とした。表4.3に評価者の属性を示す。評価者の内1名(a)は助教で、残りの6名は技術職員であった。また、すべての評価者は緊急時訓練の参加経験があった。評価者の内3名(a、d、e)はプログラミング経験者で残りの4名(b、c、f、g)はプログラミング未経験者であった。また、評価者bはセキュリティ担当の職員であった。本評価において、評価者aおよびbは手順1を、評価者c、dおよびeは同時に手順2-1を、評価者fおよびgは同時に手順2-2を行った。アンケート項目に関するインタビューは4名(a、b、e、f)に対して、評点が5でなかった項目に関してのみ行った。

表 4.3: 評価者の属性

評価者	職種	緊急時訓練参加回数	プログラミング経験
a	助教	9	有
b	技術職員	13	無
c	技術職員	11	無
d	技術職員	1	有
e	技術職員	7	有
f	技術職員	10	無
g	技術職員	12	無

## 4.3 評価結果

図4.5から図4.8に、評価者がプログラムブロックを組み立てる様子、評価者がシナリオブロックを設置する様子、評価者がシナリオブロックを撮影する様子および評価に使用したプログラムブロックを示す。



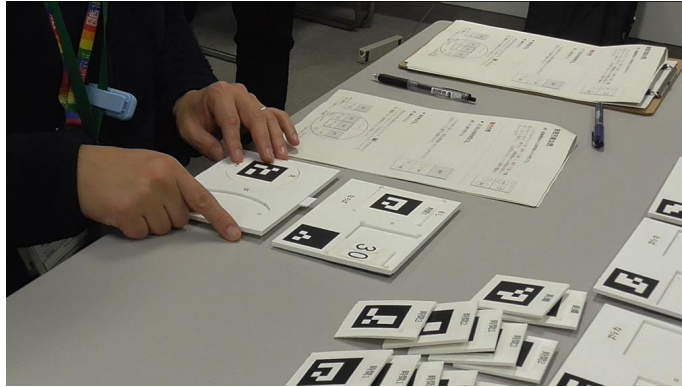


図 4.5: 評価者がプログラムブロックを組み立てる様子



図 4.6: 評価者がシナリオブロックを設置する様子



図 4.7: 評価者がシナリオブロックを撮影する様子



図 4.8: 評価に使用したプログラムブロック

評価者 a から g の容易さに関するアンケートの結果および回答理由を表 4.4 から表 4.6 に示す。また、自由作成課題で評価者が考えた AR 訓練を表 4.7 に、評価者 a と b の訓練シナリオの自由度に関するアンケートの結果および回答理由を表 4.8 および表 4.9 に示す。評価者 g の自由記述欄への回答を表 4.10 に示す。

表 4.4 における×は手順 2 を遂行した評価者にとって答えられないアンケート項目である。評価に要した時間はおおむね図 4.3 および図 4.4 に示した時間と同じであった。

表 4.4: ARTM の容易さに関するアンケートの結果

番号	アンケート項目	a	b	c	d	e	f	g
(X)-1	どのパーツブロックがどの機能を表すか理解できた。	5	4	5	5	4	5	5
(X)-2	ベースブロックにどのパーツブロックがはめ込められるか理解できた。	5	5	5	5	5	5	5
(X)-3	時間 1 を条件として設定することは容易だった。	4	5	5	5	4	4	5
(X)-4	距離を条件として設定することは容易だった。	4	4	×	×	×	4	5
(X)-5	時間 1 と時間 2 の違いを理解することは容易だった。	4	4	×	×	×	5	5
(X)-6	物 CG (火や漏水など) の再生を設定することは容易だった。	5	3	5	5	4	5	5
(X)-7	物 CG (火や漏水など) の停止を設定することは容易だった。	5	3	5	5	5	×	×
(X)-8	人物 CG (警察官、侵入者など) の動きを設定することは容易だった。	5	2	×	×	×	4	5
(X)-9	CG の表示位置を設定することは容易だった。	4	3	2	3	4	4	5
(X)-10	CG の表示向きを設定することは容易だった。	4	3	2	5	4	4	5
(X)-11	シナリオブロックを撮影することは容易だった。	4	2	3	2	2	4	3
(X)-12	想像していた通りに CG が表示された。	5	3	4	2	4	4	5
(X)-13	初めて使う人でも簡単にシステムを使用できる。	3	4	3	2	3	4	5

表 4.5: ARTM の容易さに関するアンケートの結果および回答理由 (X)-1 から (X)-10

番号	アンケート項目	評価者	評点	回答理由
(X)-1	どのパーツブロックがどの機能を表すか理解できた。	b	4	音声の表示などが小さい。
		e	4	マーカの部分も人間が理解できるマーカであれば良かった。
(X)-3	時間 1 を条件として設定することは容易だった。	a	4	時間を意識しながら作ることが難しい。
		e	4	時間 1 と時間 2 の違いが分からない。
		f	4	イメージする時間と、実際に流れている時間が異なった。シナリオをしっかり意識しなければならなかった。
(X)-4	距離を条件として設定することは容易だった。	a	4	時間と組み合わせることが難しい。
		b	4	<と=どちらを使うべきかわかりにくい。距離には<を使うと明記してほしい。
		f	4	1m は感覚的にわかるが、5m 程度になると難しい。1 から 2m のように範囲指定ができればよかった。
(X)-5	時間 1 と時間 2 の違いを理解することは容易だった。	a	4	時間 1 が理解できると難しくない。開始することが難しい。
		b	4	慣れれば理解できる。時間の種類が多くなりすぎると混乱するかもしれない。
(X)-6	物 CG の再生を設定することは容易だった。	b	3	動画やプレビューがあればわかりやすい。
(X)-7	物 CG の停止を設置することは容易だった。	b	3	(X)-6 と同じ
(X)-8	人物 CG の動きを設定することは容易だった。	b	2	課題説明資料のフローで混乱して、思った通りに設定できなかった。
		f	4	人の動きを思っているより速く設定してしまった。時間感覚の把握が難しい。
(X)-9	CG の表示位置を設定することは容易だった。	a	4	トラッキングを考慮することが難しい。
		b	3	(X)-6 と同じ
		f	4	練習があったからできた。練習するまでは想像つかなかった。
(X)-10	CG の表示向きを設定することは容易だった。	a	4	プログラムブロックに明確な表示がないため、立てかけるのか床に置くのかが分かりづらかった。
		b	3	(X)-6 と同じ
		f	4	(X)-9 と同じ

表 4.6: 容易さに関するアンケートの結果および回答理由 (X)-11 から (X)-13

番号	アンケート項目	評価者	評点	回答理由
(X)-11	シナリオブロックを撮影することは容易だった。	a	4	トラッキングを考慮することが難しい。
		b	2	トラッキングを考慮することが難しい。
		e	2	ゆっくり動くことが難しい。
(X)-12	想像していた通りにCGが表示された。	b	3	(X)-6と同じ
		e	4	プログラムブロックの内容通りには表示されたが、CGに違和感があった。
		f	4	警察官と侵入者の見た目の区別が難しい。
(X)-13	初めて使う人でも簡単にシステムを使用できる。	a	3	習得に時間がかかる。ストーリー全体を考慮することが難しい。
		b	4	練習して、マニュアルを読めば可能。
		e	3	マーカのわかりにくさとトラッキングの難しさがあった。

表 4.7: 自由作成課題で評価者が考えた AR 訓練

評価者	自由記述
a	毎年実施している訓練で、炉心からの水漏れを想定した訓練を AR で作成したい。爆発が起こってタンクから漏水が起こり、作業員が駆けつける訓練。
b	配管から水漏れを発見し、水漏れの程度から、訓練者が何をすればよいか判断し、そのアクションの正誤により訓練状況が変化する訓練。

表 4.8: 訓練シナリオの自由度に関するアンケートの結果

番号	アンケート項目	a	b
(Y)-1	現場で必要とされている訓練を作成するのに必要な物 CG (火、漏水など) を備えている。	4	2
(Y)-2	現場で必要とされている訓練を作成するのに必要な人 CG (警察官など) を備えている。	4	3
(Y)-3	現場で必要とされている訓練を作成するのに必要な物 CG の動き (再生、停止) を備えている。	5	4
(Y)-4	現場で必要とされている訓練を作成するのに必要な人 CG の動き (動く、乗り越える、飛び降りる、かがむ) を備えている。	5	3
(Y)-5	現場で必要とされている訓練を作成するのに必要な条件 (時間、ユーザーとの距離など) を備えている。	4	5

表 4.9: 訓練シナリオの自由度に関するアンケートの結果および回答理由

番号	アンケート項目	評価者	評点	回答理由
(Y)-1	現場で必要とされている訓練を作成するのに必要な物 CG を備えている。	a	4	放射線のように、実際には見えないものを表示したい。
		b	2	すべての CG に関して、大きさや勢いなど CG の程度を設定したかった。
(Y)-2	現場で必要とされている訓練を作成するのに必要な人 CG を備えている。	a	4	侵入者の数を 2 人に増やしたい。
		b	3	倒れている人などが欲しかった。実際の訓練では侵入者の特徴を警察に伝えるので、様々な特徴を持った侵入者が欲しかった。
(Y)-3	現場で必要とされている訓練を作成するのに必要な物 CG の動きを備えている。	b	4	再生停止だけでなく、徐々に大きくなるなどの段階があればよかった。
(Y)-4	現場で必要とされている訓練を作成するのに必要な人 CG の動きを備えている。	b	3	実際の訓練の際には、放射線汚染が気になり、ほふく前進ができないため、AR で模擬したい。
(Y)-5	現場で必要とされている訓練を作成するのに必要な条件を備えている。	a	4	訓練開始を自分で決められるようにしたい。システムの操作者が望むタイミングでイベントを起こしたい。

表 4.10: 評価者 g の自由記述欄への回答

評価者	自由記述
g	シナリオブロックの作成が簡単だが、撮影などの設定が難しいため、もっとスムーズにできるほうが望ましい。 必要とされる CG については、ユーザの要望により対応できると思うので、問題はないと思う。

## 4.4 結果の考察

### 4.4.1 ARTM の容易さに関する評価結果の考察

#### (1) プログラムブロックの機能に関する認知性

表 4.1 に示すように、プログラムブロックの機能に関する認知性に対応するアンケート項目は、(X)-1 および (X)-5 である。

表 4.4 の (X)-1「どのパーツブロックがどの機能を表すか理解できた。」、(X)-5「時間 1 と時間 2 の違いを理解することは容易だった。」の評点はすべての評価者において 4 以上であった。表 4.5 より、(X)-1 において評価者 b および e から「音声の表示や日本語が小さい。」、「パーツブロックの大半を占めているマークが人間にもわかるマークであれば、なおよかった。」という意見が得られた。これらはいずれもマークの大きさが日本語表記よりも大きいことについて言及している。ARTM では、その認識安定性の高さから ArUco マークを用いた。しかし ArUco マークのような人間にとっては単なる模様にはしか見えないバイナリマークではなく、人間が理解できる任意の画像や文字をマークとして設定し、自作マークを作成する技術<sup>[39]</sup>も存在する。そのような人間が理解しやすいマークはカメラと画像処理による認識が不安定であることから本研究では用いなかったが、そのようなマークが安定して認識でき本研究で利用できるようになれば、プログラムブロックの機能に関する認知性のさらなる向上が期待できる。

表 4.5 より、(X)-5 において評価者 b から「慣れれば違いを理解することは難しくないが、時間 2 だけでなく時間 3 などが増えてくるほど、イベントの時間管理が難しくなる。」という意見が得られた。これは、AR 訓練シナリオ作成において、シナリオの全体像を把握することが難しく、イベントの発生時間条件をどの程度に設定するべきかを決定することが難しいという問題について言及している。この問題は、訓練環境構築完了時に、訓練シナリオのタイムラインをタブレット PC の画面上で確認できる機能を追加することで解決できるようになると期待できる。例えば、図 4.9 に示すように、訓練シナリオのタイムラインを確認できるようになれば、各イベントの時間条件の間隔を視覚的に比較できるため、訓練シナリオを想像しやすくなる。また、時間 1 を設定したイベント、時間 2 を設定したイベントなど、各時間におけるイベントを確認できるため、時間軸が多くなっても混乱することがなくなると期待できる。

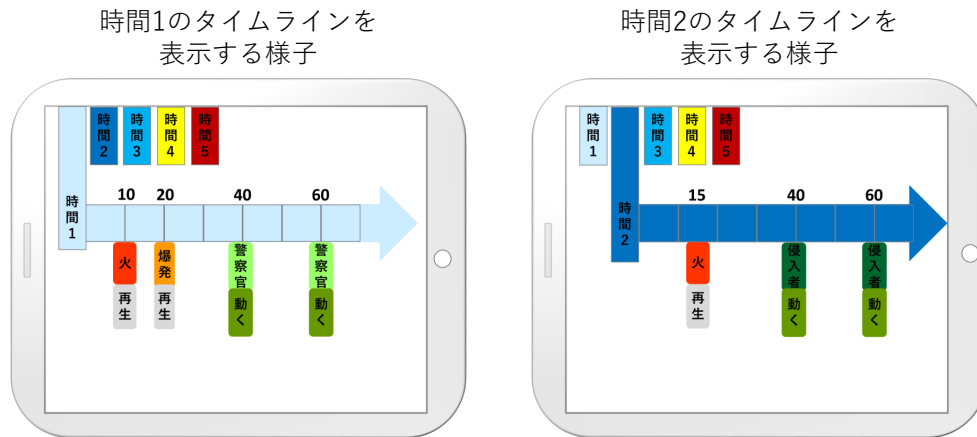


図 4.9: 訓練シナリオのタイムラインのイメージ

## (2) プログラムブロックの組み立てに関する認知性および操作性

表 4.1 に示すように、プログラムブロックの組み立てに関する認知性に対応するアンケート項目は (X)-2、認知性および操作性に対応するアンケート項目は (X)-3、(X)-4、(X)-6、(X)-7 および (X)-8 である。

表 4.4 の (X)-2 「ベースブロックにどのパーツブロックがはめ込められるかを理解できた。」の評点はすべての評価者において 5 であった。この結果により、現場の人々にとってプログラムブロックは組み立て方が理解しやすい形状をしていることがわかった。

表 4.4 の (X)-3 「時間 1 を条件として設定することは容易だった。」の評点はすべての評価者において 4 以上であった。表 4.5 より、(X)-3 において評価者 a および f から「実際に流れている時間と、数字から想像する時間が異なり、シナリオをしっかりと意識することが難しい。」、「時間を常に意識しながら作ることが難しい。」という意見が得られた。これらは (X)-1 と同じく、AR 訓練シナリオ作成において、イベントの発生時間条件をどの程度に設定するべきかを決定することが難しいという問題について言及していたため、図 4.9 のタイムラインを確認できるようになれば、解決できるようになると期待できる。

表 4.4 に示すように、(X)-4 「距離を条件として設置することは容易だった。」の評点はすべての評価者において 4 以上であった。表 4.5 より評価者 b から「演算子の < か = どちらを使うべきかわかりにくい。距離は < を使うべきであることを明記してほしい。」という意見を得た。ARTM では、例えば「ユーザがシナリオブロックを設置した位置から 2m 以内に近づいたら爆発が再生される」という条件を設定したい場合、「ユーザとの距離 < 2m ならば爆発が再生される」のように設定する。しかし「ユーザとの距

距離 = 2m ならば爆発が再生される」のように設定することもできる。ARTMでの距離計測は、数百 ms 毎に実行しており、後者はユーザとの距離がちょうど 2m となっている状態が、この距離更新時間以上続かないと条件を満たしたことになる。後者のように設定した場合、ユーザが素早く 2m 以内に近づいた場合には、爆発が再生されないことが起こり得る。このようなシステムのプログラムの仕様まで考慮してシナリオブロックを組み立てることをプログラミング初心者には要求することは難しい。ARTMでは距離パーツブロックと時間パーツブロックの形状は同じであったが、形状を異なるものとして、距離パーツブロックは<と>のベースブロックだけにしか使えないという設計に変更すれば、この問題は解決できると考えられる。

また、表 4.5 より、評価者 f から「5m など大きい数字になるほど、ユーザが想像する距離と実際の距離とのギャップが大きくなってしまふ。1~2m など範囲指定できたらよい。」という意見を得た。評価者の提案の通り、距離条件を範囲指定にすることも可能であるが、それによってユーザの想像する距離と実際の距離とのギャップを埋めることはできない。そのため、図 4.10 に示すように、距離条件を設定したシナリオブロックを撮影した時に、距離条件の範囲をタブレット PC 上で確認できるように AR でハイライト表示するという解決策が有効だと考えられる。これによって、距離条件の範囲を視覚的に確認でき、ユーザの想像する距離と実際の距離とのギャップを埋めることができるようになると考えられる。

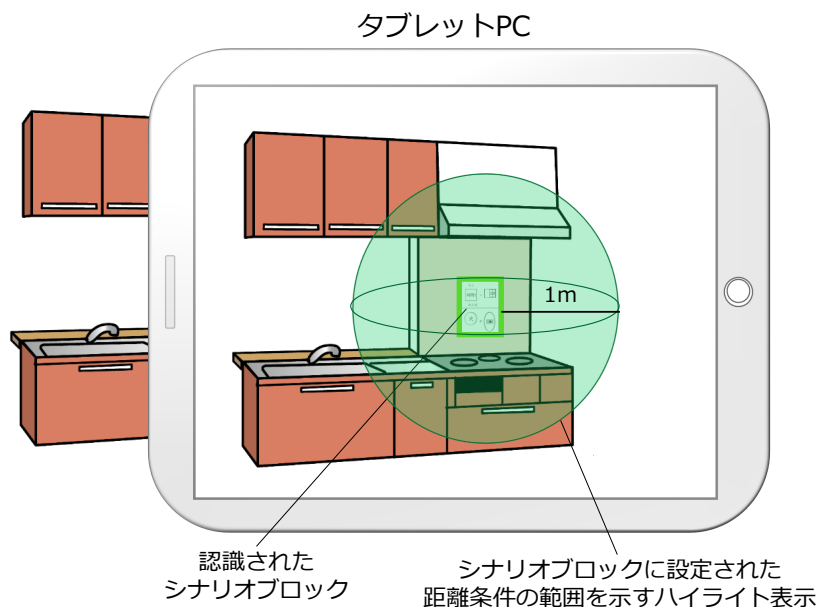


図 4.10: 距離条件の範囲をハイライト表示するイメージ



表 4.4 に示すように、(X)-6「物 CG（火や漏水など）の再生を設定することは容易だった。」、(X)-7「物 CG（火や漏水など）の停止を設定することは容易だった。」の評点は、評価者 b において 3、そのほかの評価者において 4 以上であった。表 4.5 より、(X)-6 および (X)-7 の両方で評価者 b から「体験するまでどのような CG が表示されるかわからないため、動画やプレビューのようなものがあつたほうがよかった。」という意見を得た。この指摘に関して、図 4.11 に示すように、シナリオブロック撮影時にシナリオブロックに設定されている動作を簡易的に AR 表示するプレビュー機能のようなものを追加すれば、解決できるようになると期待できる。

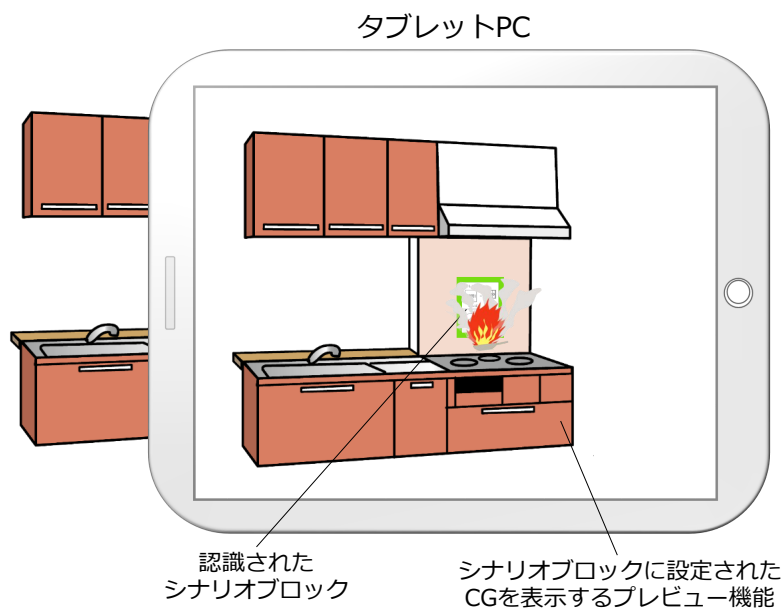


図 4.11: プレビュー機能のイメージ

(X)-8「人物 CG（警察官、侵入者など）の動きを設定することは容易だった。」の評点は評価者 b を除いて 4 以上であった。評価者 b が 2 と回答した理由は「課題説明資料の図で混乱して、課題において思った通りに設定できなかったから。」であった。評価者 b は図 D.2 におけるタイムラインのイメージ図をみて、侵入者が逃げる様子を模擬する際に、侵入者の動きを評価者 b 自身が想定していたよりも遅く設定してしまった。また、評価者 f が 4 と回答した理由は「人物の動きを速く設定しすぎたから。時間感覚を把握することが難しい。」であった。これらは、人物の動きを時間で管理することが難しいという問題に言及している。この問題は、(X)-6 と同様に、図 4.11 に示したように、シナリオブロック撮影時にシナリオブロックに設定されている動作を簡易的に AR 表示するプレビュー機能のようなものを追加すれば、AR 訓練体験前に人の動きを

確認できるため、解決できると期待される。

### (3) プログラムブロックの設置に関する認知性および操作性

(X)-9「CGの表示位置を設定することは容易だった。」の評点は評価者b、cおよびdは2もしくは3、そのほかは4以上であった。表4.5より、評価者aが4と回答した理由は「トラッキングを意識して設置することが難しかったから。」であった。本評価ではトラッキングに失敗したことにより、シナリオブロックの設置位置を調整したことがあった。この問題はプログラムブロックの入力方法に関する操作性における問題であるため、(4)で解決策を検討する。また、(X)-10「CGの表示向きを設定することは容易だった。」のアンケート結果は評価者bおよびcを除いて、4以上であった。(X)-9および(X)-10において評価者bが3と回答した理由は(X)-6と同じく、事前にCGがどのような向きやサイズ感で現れるのかがわからないことに言及していた。この問題は、(X)-6と同様に、シナリオブロック撮影時にシナリオブロックに設定されている動作を簡易的にAR表示するプレビュー機能のようなものを追加することで解決されると期待できる。

### (4) プログラムブロックの入力方法に関する操作性

(X)-11「シナリオブロックを撮影することは容易だった。」の評点は2が最も多かった。表4.6より、評価者a、b、eが4、2、2と回答した理由はいずれもトラッキングにたびたび失敗してしまい、難しいというものだった。評価者eはトラッキングの難しさの中でも、トラッキング中に許容されるスピードがわからないことを言及していた。この結果から、プログラムブロックの入力方法に関する操作性は、難しいことが分かった。

しかし、これらはプログラムブロックの認識ではなくトラッキングの難しさだけといえる。トラッキングの性能は日々向上しているため、性能が向上したトラッキング方法を用いることで解決されると期待できる。

### (5) ARTM全体の認知性および操作性

(X)-12「想像していた通りにCGが表示された。」の評点は評価者bは3、評価者dは2、そのほかは4以上であった。また(X)-13「初めて使う人でも簡単にシステムを使用できる。」の評点は評価者a、cおよびeは3、評価者dは2、そのほかは4以上であった。(X)-13において、表4.6より、評価者aは習得に時間がかかることを言及していた。また、評価者bは、はじめは難しいが練習とマニュアルがあれば習得できると言及していた。ARTMの使用は容易であるとは言い切れないが、これまで、AR訓練環境を構築するためにプログラミング言語を1から学ばなければならなかったことと比べると、30分程度で使用方法を習得できることから、従来のAR訓練環境の構築

方法と比べると、ARTM を用いることにより大幅に構築が容易になったと言える。また(1)から(4)で検討したように、システムの根本的な修正は必要なく、複数の機能を新たに加えることで、ARTM の使用をより容易にできることが分かった。

#### 4.4.2 訓練シナリオの自由度に関する評価結果の考察

##### (1)CG の種類

表 4.8 の (Y)-1 「現場で必要とされている訓練を作成するのに必要な物 CG を備えている。」の評点は、評価者 a は 4、評価者 b は 2 であった。表 4.9 より、評価者 a の回答理由は、「放射線のように実際には見えないものを表示したいから。」であった。3.3.1 項のプログラムブロックで実現する訓練要素の検討の際には、重要施設において実施されている訓練を参考にしたが、回答理由のように AR だからこそ実現できる訓練要素も求められていることが分かった。放射線は、疑似的な放射線を表す CG を作成し、火の CG と同様に設定して表示可能にできるため、放射線の CG を表すパーツブロックを追加することで対応できる。

評価者 b の回答理由は、「すべての CG に関して、大きさや勢いなどの程度を設定したかったから。」であった。本システムでは、火や漏水を設定したシナリオブロックを複数並べて設置することで、大きな火や漏水のように見せることができるが、小さくすることや、勢いを設定することはできない。この問題に対しては、大きさや勢いを連続的に変化させられるようにする必要はなく、段階的に調整できるだけで十分であると考えられる。そこで、大きさや勢いが異なる CG を表すパーツブロックを複数追加で用意することにより対応できる。

(Y)-2 「現場で必要とされている訓練を作成するのに必要な人 CG を備えている。」の評点は、評価者 a は 4、評価者 b は 3 であった。評価者 a および b の回答理由は、侵入者の数および種類を増やしたいからであった。この問題も、(Y)-1 と同様に、侵入者のパーツブロックの種類を追加することで対応できる。表 4.9 より、評価者 b の回答理由の「倒れている人なども欲しかったから。」などは人 CG の動きの問題であるため、その解決策は (2) で述べる。

##### (2)CG の動きの種類

(Y)-3 「現場で必要とされている訓練を作成するのに必要な物 CG の動きを備えている。」の評点は、評価者 a は 5、評価者 b は 4 であった。評価者 b の回答理由は、物 CG の勢いに言及しており、(Y)-1 で述べた、大きさや勢いが異なる CG を表すパーツブロックを複数追加で用意することにより対応できると考えられる。

(Y)-4「現場で必要とされている訓練を作成するのに必要な物 CG の動きを備えている。」の評点は、評価者 a は 5、評価者 b は 3 であった。評価者 b の回答理由は、「倒れている人や、ほふく前進などの動きを表現したいから。」であった。この問題は、(Y)-1 と同様に、人の動きのパーツブロックの種類を追加することで対応できる。

### (3) 条件の種類

(Y)-5「現場で必要とされている訓練を作成するのに必要な条件を備えている。」の評点は、評価者 a は 4、評価者 b は 5 であった。評価者 a の回答理由は、「システム操縦者が訓練体験者を観察しながら、望むタイミングでイベントを起こしたいから。」であった。この要望に対する解決策としては、訓練生が使用しているタブレット PC とは別のシステム操作者用のタブレット PC を用意し、その画面上に (X)-3 で述べた解決策としてのタイムラインを表示し、訓練実施中にシステム操作者が、画面をタップすることで自由なタイミングで予め設定したイベントを発生させることができるようにするなどの方法が考えられる。

(1) から (3) の結果から、ARTM で作成できる訓練シナリオの自由度は高いとは言いきれないが、放射線、侵入者の種類、人が倒れる、ほふく前進するなどの新しいプログラムブロックの種類や、火や漏水の CG の大きさや勢いが異なるプログラムブロックの種類を追加することによって、ARTM で作成できる訓練シナリオの自由度は向上すると考えられる。また、システム操作者が画面をタップすることで自由なタイミングで予め設定したイベントを発生させることができるようにすることによっても、訓練シナリオの自由度は向上すると考えられる。

## 第 5 章 結論

本研究では、重要施設における安全性に対するソフト面からの対策として AR 訓練に注目し、AR 訓練環境の構築に必要な専門知識を持っていない人でも容易に AR 訓練環境を構築可能にすることを目標とした、AR 訓練環境構築システム (ARTM) を開発した。ARTM の目標を達成するために、プログラミングの高度なスキルを持っていない人でも容易に使用できる新しい実世界指向プログラミング言語「プログラムブロック」を開発した。また、ARTM の目標である、容易に理解および操作ができること、現場で求められている水準の複雑さの訓練シナリオを作成できることを評価するため、防災訓練や不法事案対策訓練を定期的に行っている重要施設の職員に使用してもらい評価した。

第 2 章では、研究の背景として、AR 訓練における課題点を述べ、次に AR 訓練環境構築に関する既往研究について述べた。また、本研究の目的が、実世界指向プログラミングを用いた ARTM の開発であることを述べた。

第 3 章では、ARTM の設計と実装について述べた。はじめにシステムの目標を述べ、目標を達成するために開発したプログラミング言語、プログラムブロックについて述べた。次に、ARTM を構成する 2 つのサブシステム、訓練環境構築サブシステムと訓練環境体験サブシステムの設計について述べた。その後、設計に基づいた ARTM の実装について述べた。

第 4 章では、第 3 章で開発した ARTM 使用時の理解および操作の容易さと、ARTM で作成できる訓練シナリオの自由度を、防災訓練や不法事案対策訓練を定期的に行っている重要施設の職員に使用してもらうことにより評価した。本評価により、ARTM の使用は容易であるとは言い切れないが、従来の構築方法と比べると、AR 訓練環境を容易かつ短時間で構築可能になったことが分かった。一方、プログラムブロックには、ユーザは一目見た時に、それが何を表しているのかを瞬時に理解することが難しいという問題点があることが分かった。今後、人間が理解できる文字や画像をマーカとして用いることによって、プログラムブロックの認知性を向上させる必要があると考えられる。また、訓練シナリオのタイムラインをタブレット PC の画面上で確認できる機能を追加することによって、プログラムブロックの機能および組み立ての認知性を向上できると期待されることが分かった。他にも、プログラムブロック組み立ての認

知性を向上させるためには、距離パーツブロックと時間パーツブロックの形状を異なるものとするのが考えられる。さらに、距離条件を設定したシナリオブロックの撮影時に、ユーザが距離条件の範囲をタブレット PC 上で確認できるように AR でハイライト表示する機能や、シナリオブロック撮影時にシナリオブロックに設定されている動作を簡易的に AR 表示するプレビュー機能を追加することによって、プログラムブロックの組み立ておよび設置に関する認知性を向上できると考えられる。また、現状の ARTM を用いて作成できる訓練シナリオの自由度は十分高いとは言い切れないが、複数の追加機能および、プログラムブロックの種類を増やすことによって現場で求められている AR 訓練を作成できるようになると分かった。その他の改善案として、システム操作者用のタブレット PC を用意し、タイムラインを表示し、訓練実施中にシステム操作者が、画面をタップすることで自由なタイミングで予め設定したイベントを発生させることができるようにする機能を追加することが考えられる。

本研究では、現場で容易に AR 訓練環境を構築できるシステムを開発するために、実世界指向プログラミングとビジュアルプログラミングを参考にしたプログラミング言語、プログラムブロックを開発した。プログラムブロックは実世界指向であるため、現場で直感的に訓練シナリオを設定できたが、例えば人の動きをはじめとするシナリオの細部は数値で設定する必要がある、それらは実世界指向で扱える対象ではなく、未だ、情報世界の中での方が扱いやすいままとってしまった。残念ながら、現在の技術では訓練シナリオの全てを実世界指向で記述することは難しかった。しかし今後の技術の発展に従い、例えばシナリオ内における人の動きをさらに直感的に設定可能にするために、実際の人動きをキャプチャして設定に使用可能にするなど、より実世界指向な訓練シナリオの記述方法を用いることができるようになると思われる。

本研究により、AR 訓練環境の構築における実世界指向プログラミングの有用性を示すことができ、今後の AR 訓練環境の構築構築を容易にする実用的なシステムの実現に向けて、有用な知見を提供できたのであれば幸甚の至りである。

## 謝 辞

研究生活において、研究会などでの適切な助言やご指導頂き、下田宏教授に心より感謝いたします。研究だけでなく、就職活動に関する助言や人生観など、大変励みになりました。

研究の進め方から、システムの構築、論文執筆、うどん作成などありとあらゆる面で数えきれないほどご指導とご支援をいただいた石井裕剛准教授に心より感謝いたします。なにか新しい研究テーマを考えて、システムを作るということがしたくて進学したのですが、それが達成できたのは、無知な自分に丁寧にご指導して頂いたおかげです。本当に感謝しております。

事務手続きなど、研究生活全般で様々なご支援をいただいた普照郁美秘書に深く感謝申し上げます。よく話しかけて下さって楽しかったです。

本システムの評価を実施するにあたり、ご協力頂きました、高橋助教に深く感謝申し上げます。また、評価に参加して頂き、研究に関する貴重なご意見を頂きました、複合原子力科学研究所の職員の皆様に深く感謝申し上げます。

同じ研究グループで、ミーティングへの参加や、システム開発に関する助言をして頂いた、エネルギー情報学研究室の博士課程の原園さんに深く感謝申し上げます。

また、システムの評価の手伝いや、論文執筆の際に全面的にサポートして頂いた、エネルギー科学研究科修士1回生の大本君、宮崎君に深くお礼申し上げます。

研究のみならず、学生生活全般で支えて頂き、楽しく有意義な研究室生活を過ごせるようにして下さい、エネルギー情報学研究室の博士課程の上田さん、修士2回生の木村君、久留島さん、竹川さん、東山君、三木君、Piaさん、修士1回生の坂本佳樹君、坂本龍平君、高島さん、Dingming君、工学部電気電子工学科4回生の魚谷君、山脇さん、湯村君、エネルギー情報学研究室のOBの方々に深く感謝申し上げます。

最後に、支えてくださった家族や友人にお礼申し上げます。

## 参考文献

- [1] 電気事業連合会: 原子力防災体制の整備 原子力防災対策—電気事業連合会, <https://www.fepc.or.jp/nuclear/safety/bousai/kyouka/index.html> Accessed January 20, (2020).
- [2] 関西電力: 原子力発電所における津波・浸水対策—津波に備える—あくなき安全性の追求 | 原子力発電について | エネルギー | 事業概要 | 関西電力, [https://www.kepcoco.jp/energy\\_supply/energy/nuclear\\_power/anzenkakuho/taisaku/various\\_risk/tsunami.html](https://www.kepcoco.jp/energy_supply/energy/nuclear_power/anzenkakuho/taisaku/various_risk/tsunami.html) Accessed January 20, (2020).
- [3] 関西電力: 原子力発電所における火山・竜巻・森林火災等の自然現象への対策 [関西電力], [https://www.kepcoco.jp/energy\\_supply/energy/nuclear\\_power/anzenkakuho/taisaku/various\\_risk/shizensaigai.html](https://www.kepcoco.jp/energy_supply/energy/nuclear_power/anzenkakuho/taisaku/various_risk/shizensaigai.html) Accessed January 20, (2020).
- [4] 関西電力: 安全への取り組み—私たちの取り組み—火力発電について—エネルギー—事業概要—関西電力, [https://www.kepcoco.jp/energy\\_supply/energy/thermal\\_power/ourapproach/index.html](https://www.kepcoco.jp/energy_supply/energy/thermal_power/ourapproach/index.html) Accessed January 18, (2020).
- [5] 内閣府: 原子力防災訓練・研修: 原子力防災-内閣府 [https://www8.cao.go.jp/genshiryoku\\_bousai/kunren/kunren.html](https://www8.cao.go.jp/genshiryoku_bousai/kunren/kunren.html) Accessed January 18, (2020).
- [6] 警察庁: 警察庁 警備情勢を顧みて 第3章 国際テロ情勢, [https://www.npa.go.jp/archive/keibi/syouten/syouten284/pdf/03\\_16-19P.pdf](https://www.npa.go.jp/archive/keibi/syouten/syouten284/pdf/03_16-19P.pdf) Accessed January 18, (2020).
- [7] 内閣府: 企業の事業継続マネジメントにおける連携訓練の手引き, [http://www.bousai.go.jp/kyoiku/kigyou/keizoku/pdf/tebiki13\\_03.pdf](http://www.bousai.go.jp/kyoiku/kigyou/keizoku/pdf/tebiki13_03.pdf) Accessed January 19, (2020).



- [8] 日本経済新聞: 整備訓練に AR エアバス・JAL の異色タッグで開発,  
<https://www.nikkei.com/article/DGXMZO23447800U7A111C1XA0000/>,  
Accessed January 18, (2020)
- [9] Lovreglio, R.: A Review of Augmented Reality Applications for Building Evacuation, 17th International Conference on Computing in Civil and Building Engineering (2018).
- [10] インセプト: AR(拡張現実感)とは IT用語辞典- e-Words, <http://e-words.jp/w/AR.html> Accessed January 18, (2020).
- [11] 原園 友規: 3次元再構成モデル作成のための拡張現実感を用いた環境撮影支援システムの開発, エネルギー科学研究科, 京都大学 (2019).
- [12] 板宮 朋基, 平川 俊貴, 新聞 幹久, 近田 静也, 西部 連, 富田 祐規: バーチャル火点を現実空間の任意の場所に配置できる AR 消火訓練アプリの開発, 日本バーチャルリアリティ学会論文集 (2019年9月)5D-03(2019).
- [13] 入江 祐生, 光原 弘幸, 獅々堀 正幹: AR とスマートフォン HMD を用いたインタラクティブ避難訓練システムの試作, 教育システム情報学会, 第 42 回全国大会 pp.311-312(2017).
- [14] Hiroyuki Mitsuhashi, Masami Shishibori, Junya Kawai, Keisuke Iguchi: Game-Based Evacuation Drills Using Simple Augmented Reality, IEEE 16th International Conference on Advanced Learning Technologies, pp.133-137(2016).
- [15] iDEACLOUD: 防災 VR/AR <https://bousai-vr.com/> , Accessed January 26, (2020).
- [16] Tomoki Itamiya, Hideaki Tohara, Yohei Nasuda: Augmented Reality Floods and Smoke Smartphone App Disaster Scope utilizing Real-time Occlusion, IEEE Conference on Virtual Reality and 3D User Interfaces (VR), p.1397 ,(2019).
- [17] 増井 俊之: 実世界志向プログラミング, 第 40 回プログラミングシンポジウム報告集, pp19-25(1999).

- [18] Mitchel Resnick, John Maloney, Andres Monroyhernandez, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silverman, Yasmin Kafai, Scratch: Programing for all, Communications of the acm, Vol.52, No.11, pp.60-67(2009)
- [19] Unity Technologies, <https://unity.com/> Accessed February 10, (2020).
- [20] ソニー, MESH : 小さな便利を形にできる、ブロック形状の電子ブロック | ソニー, <https://meshprj.com/jp/> , Accessed January 18, (2020).
- [21] 林 理: 火災時の避難行動のシミュレーション, 行動計量学, 第 16 卷, 第 1 号, pp.8-20(1988).
- [22] Scratch, スクラッチのブロック一覧 — スクラッチのプログラミング入門, <https://scratch-howto.com/reference/index.php>, スクラッチのブロック一覧 — スクラッチのプログラミング入門.
- [23] Anna Syberfeldt, Oscar Danielsson, Patrik Gustavsson, Augmented reality smart glasses in the smart factory:Product evaluation guidelines and review of available products, IEEE access, Vol.5, pp.9118-9130(2017).
- [24] Toshikazu Karitsuka, Kosuke Sato: A Wearable Mixed Reality with an On-board Projector, Proceedings og the Second IEEE and ACM International Symposium on Mixed and Augmented Reality(ISMAR '03)(2003).
- [25] Huidong Bai, Mark Billinghurst, Lei Gao, Jihad El-Sana: Markerless 3D Gesture-based Interaction for Handheld Augmented Reality Interfaces, Extended Abstracts of the IEEE International Symposium on Mixed and Augmented Reality 2013(2013).
- [26] 青山 周平: 拡張現実感を用いた仮置・運搬作業シミュレーションシステムの開発, エネルギー科学研究科, 京都大学 (2011).
- [27] Bruce Thomas, Victor Demczuk, Wayne Piekarski, David Hepworth, Bernard Gunther: A Wearable Computer System with Augmented Reality to Support Terrestrial Navigation, 2ND. International Symposium on Wearable Computers, pp.166-167(1998)

- [28] Joseph Newman, David Ingram, Andy Hopper: Augmented reality in a wide area sentient environment, Proc. IEEE and ACM International Symposium on Augmented Reality, pp.77-86(2001)
- [29] Bolan Jiang, Ulrich Neumann: Extendible Tracking by Line Auto-Calibration, Proc. IEEE and ACM International Symposium on Augmented Reality, pp.97-103(2001)
- [30] Boost C++ Libraries, <https://www.boost.org>, Accessed January 18, (2020).
- [31] OpenNI2 library, <https://structure.io/openni>, Accessed January 18, (2020).
- [32] OpenCV library, <https://opencv.org>, Accessed January 18, (2020).
- [33] Francisco J.Romero-Ramirez, Rafael Muñoz-Salinas, Rafael Medina-Carnicer:Speeded up detection of squared fiducial markers, Image and Vision Computing, Vol.76, pp.38-47(2018).
- [34] S. Garrido-Jurado, R. Muñoz Salinas, F.J. Madrid-Cuevas, R. Medina-Carnicer:Generation of fiducial marker dictionaries using mixed integer linear programming, Pattern Recognition:51, pp.481-491(2016).
- [35] Unity Asset Store <https://assetstore.unity.com/>, Accessed February 1, (2020).
- [36] mixamo <https://www.mixamo.com/>, Accessed February 1, (2020).
- [37] tesseract-ocr library, <https://github.com/tesseract-ocr/>, Accessed January 18, (2020)
- [38] 黒須 正明: “ヒューマンインタフェース” 田村博編, オーム社出版局, pp.426-429(1998).
- [39] Philip Lamb: ARToolKit ホームページ <http://www.hitl.washington.edu/artoolkit/>, Accessed January 18, (2020).

# 付録 A プログラムブロックで作成可能なシナリオ

プログラムブロックで作成可能なシナリオの例を図 A.1 から図 D.11 に示す。

## 表現可能な例

例) 訓練開始から30秒後に火を再生する。

シナリオブロックは、**条件** (上部) と、**動作** (下部) で構成されています。例では、「訓練開始からの時間が30秒ならば」という**条件**と、「火が再生される」という**動作**で構成されています。



## 結果の例

◆ 2つ以上のCGを表示する。

例) 火と煙を再生する。

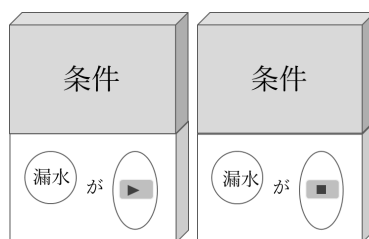
1つの条件に対して、結果は複数個つけることができます。右図のように下に動作を連結できます。



◆ CGを消す。

例) 漏水を止める。

再生済みのCGは停止することもできます。その場合、再生したシナリオブロックと並べて設置してください。



5

図 A.1: プログラムブロックで作成可能なシナリオ 1

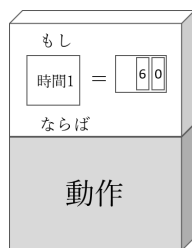
## 表現可能な例

### 条件部分の例

#### ◆ 時間1を条件に設定する

時間ブロックを左上にはめ込む場合、右上にはめ込んだ数字ブロックの単位は**秒**になります。数字は4桁まで設定できます(0~9999)。時間1ブロックは訓練開始からの時間を表し、時間2以降のブロックは任意のタイミングで開始することができます。時間2以降については次ページで説明いたします。

例) 訓練開始から60秒後



例) 訓練開始から180秒後



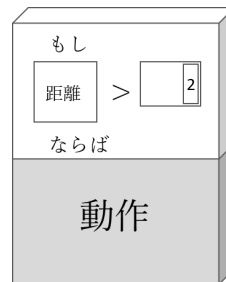
#### ◆ ユーザーとの距離を条件に設定する

距離ブロックを左上にはめ込む場合、右上にはめ込んだ数字ブロックの単位は**メートル**になります。時間条件と同じく数字は4桁まで設定できます。(0~9999)

例) ユーザーが1メートル以内に近づいたら、爆発する。



例) ユーザーが2メートル以上離れたら、~する。



6

図 A.2: プログラムブロックで作成可能なシナリオ 2

## 表現可能な例

### ◆ 時間2を条件にして、CGを表示する。

時間2以降は任意のタイミングで開始することができます。

例) Step1: ユーザーが1メートル以内に近づいたら爆発し、時間2を開始する。  
Step2: 時間2が開始して20秒後、火災が発生する。

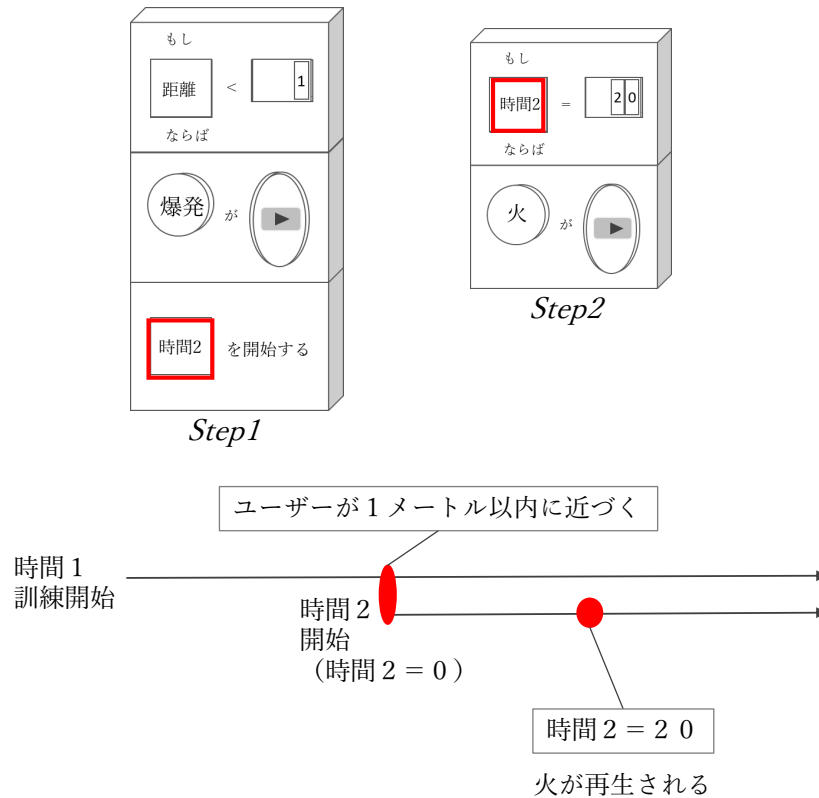


図 A.3: プログラムブロックで作成可能なシナリオ3

## 表現可能な例：人物CG (1)

◆ 時間を条件に設定し、人物CGの動きを設定する。

例) 廊下を侵入者が歩く

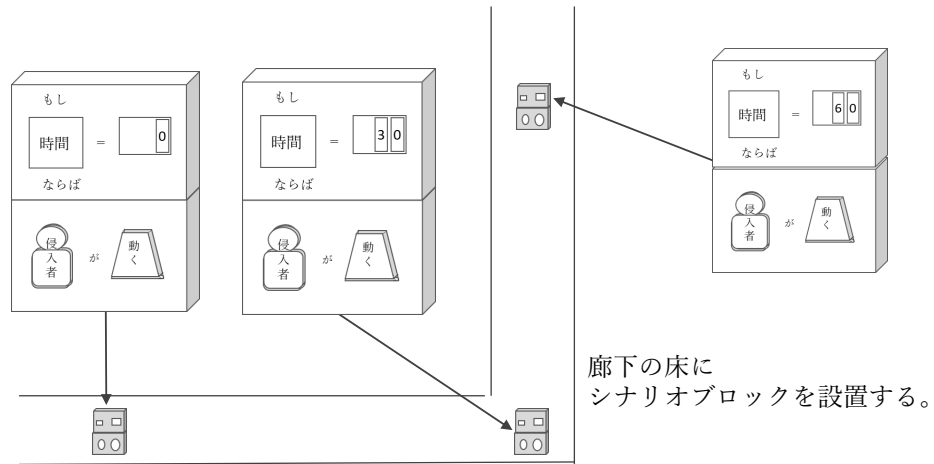


図 廊下の平面図

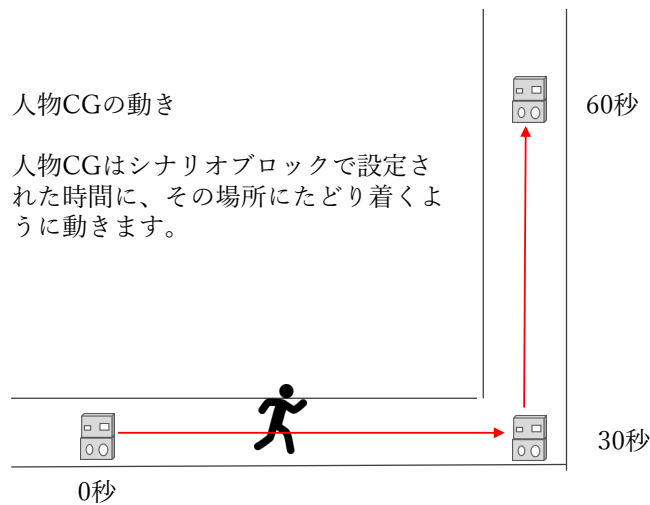


図 廊下の平面図

8

図 A.4: プログラムブロックで作成可能なシナリオ 4

## 表現可能な例：人物CG(2)

例) 侵入者がフェンスを飛び越える動作をする

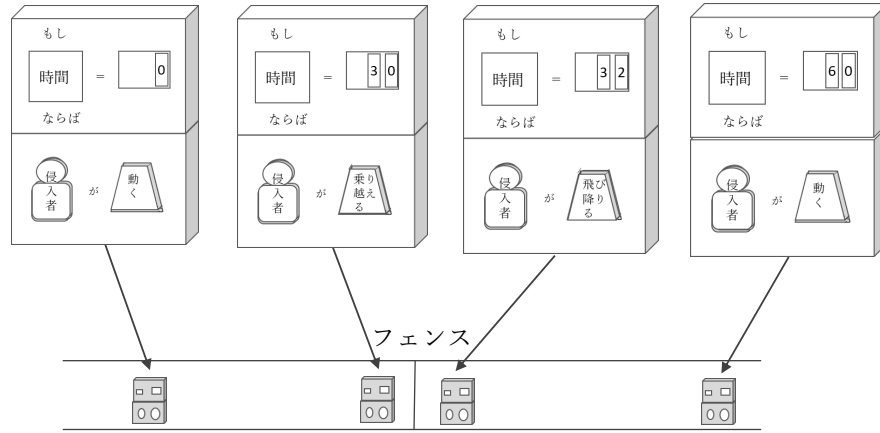


図 道の平面図

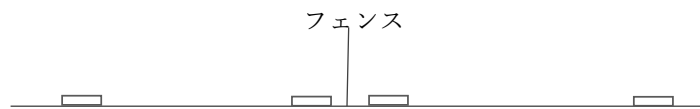


図 道の立面図

人物CGの動き

人物CGはシナリオブロックで設定された時間に、その場所にたどり着くように動き、シナリオブロックの場所で動作をします。

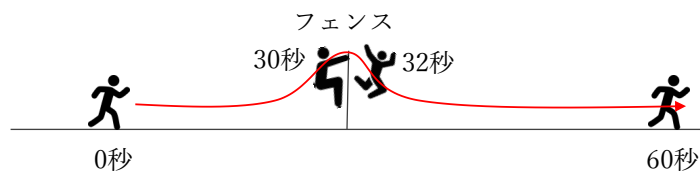


図 道の立面図

9

図 A.5: プログラムブロックで作成可能なシナリオ5



## 表現可能な例：人物CG(3)

例) 侵入者がものをあさる動作をする

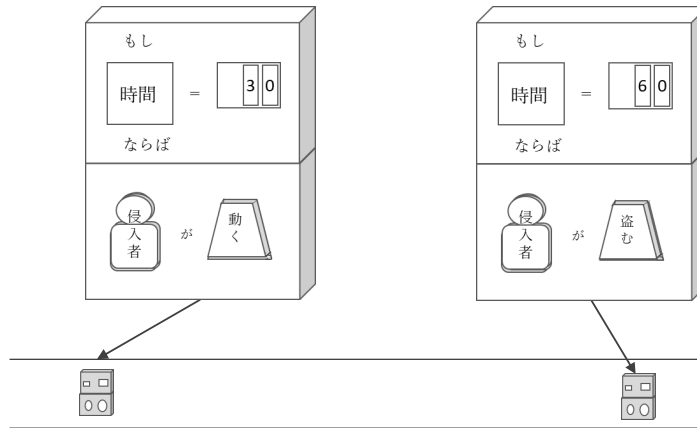


図 廊下の平面図

人物CGの動き

人物CGはシナリオブロックで設定された時間に、その場所にたどり着くように動き、シナリオブロックの場所で動作をします。

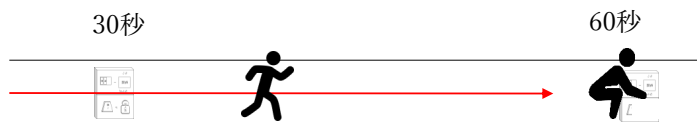


図 廊下の平面図

10

図 A.6: プログラムブロックで作成可能なシナリオ 6

## 表現可能な例：人物CG(4)

### ◆ 委間2以降を条件にして動く

- 例) Step1:ユーザーが1メートル以内に近づいたら通報し、時間2を開始する  
 Step2: Step1の180秒後 (時間2=180) に警察官が到着する  
 Step3: 20秒かけて (時間2=200まで) 部屋に向かって動く

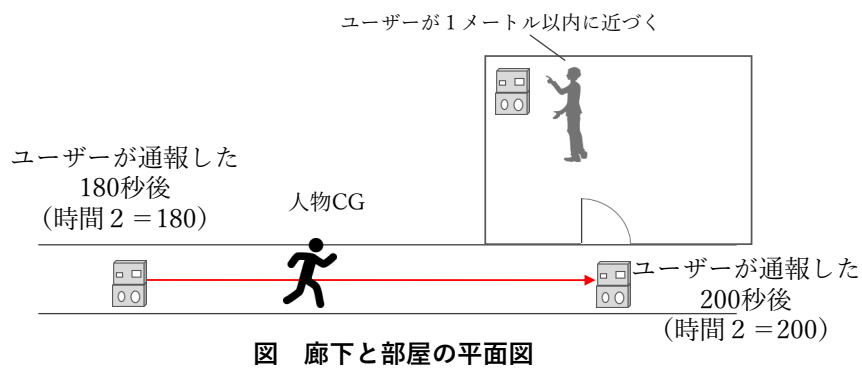
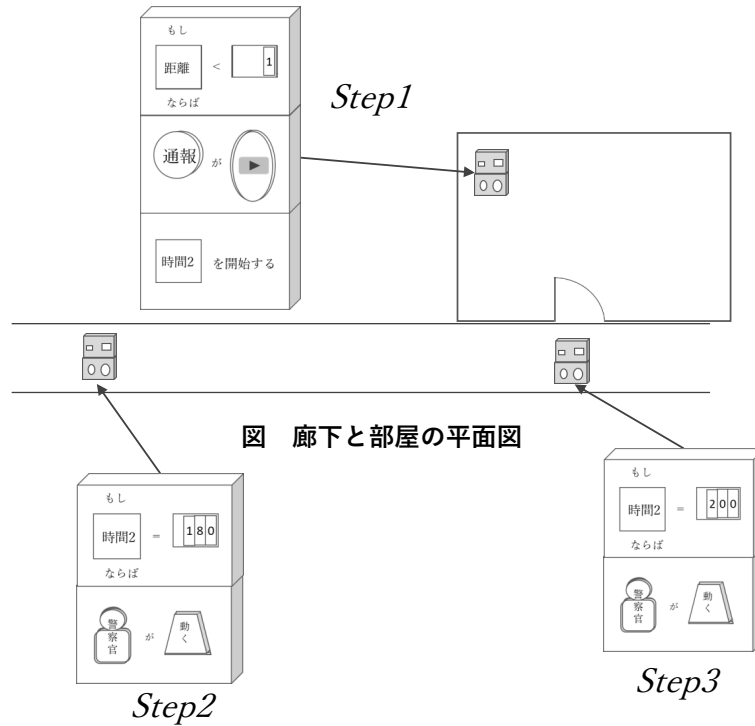


図 A.7: プログラムブロックで作成可能なシナリオ 7

## 付録 B マーカレストラッキング

特徴点ベースのマーカレストラッキングでは、図 B.1 に処理の流れを示すように、カメラから得られる画像を読み込み、過去の入力画像との間の相対的なカメラ姿勢を求めることで、特定の座標系を基準としたカメラ姿勢を順に推定する。このとき、基準となる座標系をワールド座標系と呼ぶ。画像取得時のカメラ姿勢と画像局所特徴が既知の画像  $I_P$  と、新しく撮影された画像  $I_C$  との間の相対的なカメラ姿勢を求める際には、まず  $I_C$  の画像局所特徴を計算する。次に、 $I_P$  上の特徴点のうち、 $I_C$  上の特徴点の特徴量と似た特徴量を持つ特徴点を求める。この処理を特徴点マッチングと呼ぶ。2枚の画像における特徴点検出の結果の例を図 B.2 に、特徴点マッチングの結果の例を図 B.3 に示す。その後、特徴点マッチングの結果を利用して、RANSAC などのアルゴリズムを用いて誤対応を排除しながら Perspective-n-Point (PnP) 問題を解くことで、 $I_P$  取得時のカメラ姿勢を基準とした  $I_C$  取得時のカメラ姿勢を求める。その後、ワールド座標系を基準とした  $I_P$  取得時のカメラ姿勢を用いて、ワールド座標系を基準とした  $I_C$  取得時のカメラ姿勢を推定する。全画像取得時のカメラ姿勢を推定した後、Bundle Adjustment を用いて推定誤差を低減させる処理を行う場合もある。

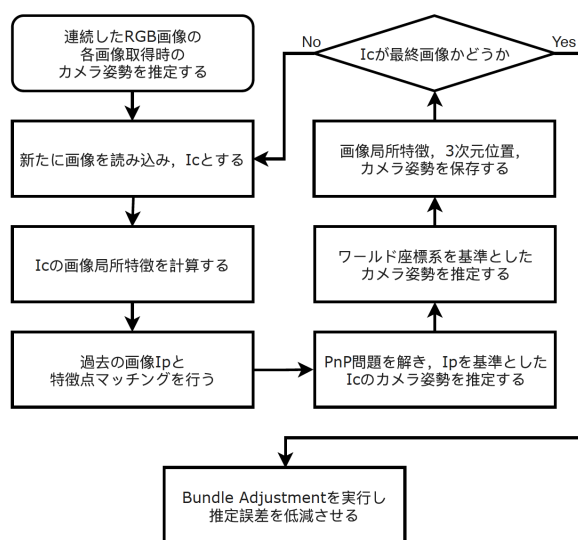


図 B.1: 特徴点ベースのマーカレストラッキングの処理の流れ



図 B.2: 2枚の画像における特徴点検出の結果の例

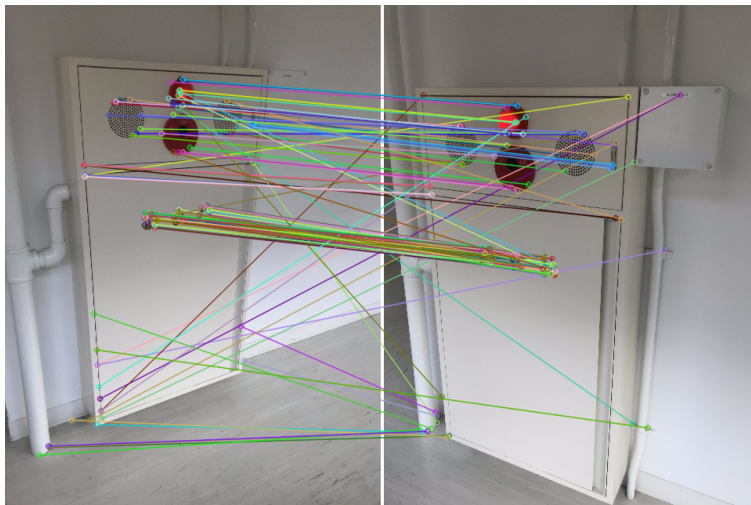


図 B.3: 2枚の画像における特徴点マッチングの結果の例

## 付録 C シナリオファイルのフォーマット

シナリオファイルには、訓練環境体験時サブシステムで用いる、3Dモデルを配置する命令、時間条件に関する命令、距離条件に関する命令、の大きく分けて3種類の命令を記すようにした。3Dモデルを配置する命令のフォーマットとその例を図C.1に、時間条件に関する命令のフォーマットとその例を図C.2から図C.4に、距離条件に関する命令のフォーマットとその例を図C.5および図C.6に示す。

### フォーマット

add, CGの種類, X座標, Y座標, Z座標, クォータニオンW, クォータニオンX, クォータニオンY, クォータニオンZ, オブジェクトID

### 例

add, fire, 10, 1.5, 5, 1.0, 0.0, 0.0, 0.0, 3

図 C.1: 3D モデル配置のフォーマットと例

### フォーマット

time, 時間の種類, 数字 (ms) , CGの動きの種類, オブジェクトID, ボードナンバー

### 例

time, 1, 10000, on, 2, 240

図 C.2: 時間条件のフォーマットと例

### フォーマット

time, 時間の種類, 数字 (ms) , sound, 音の種類, ボードナンバー

### 例

time, 1, 10000, sound, explosion, 243

図 C.3: 時間条件で効果音を再生する際のフォーマットと例

#### フォーマット

time, 時間の種類, 数字 (ms) , move, オブジェクトID, X座標, Y座標, Z座標, クォータニオンW, クォータニオンX, クォータニオンY, クォータニオンZ, X座標, Y座標, Z座標, クォータニオンW', クォータニオンX', クォータニオンY', クォータニオンZ',

#### 例

time, 1, 20000, move, 1, 5, 0, 0, 1, 0, 0, 0, 10, 0, 0, 1, 0, 0, 0

図 C.4: 人物が移動する際のフォーマットと例

#### フォーマット

location, 数字 (mm) , 符号(=, >, <), X座標, Y座標, Z座標, CGの動きの種類, オブジェクトID, ボードナンバー

#### 例

location, 1000, 2, 10, 1.5, 5, off, 3, 243

図 C.5: 距離条件のフォーマットと例

#### フォーマット

location, 数字 (mm) , 符号(=, >, <), X座標, Y座標, Z座標, 音の種類, sound, ボードナンバー

#### 例

location, 1000, 2, 10, 1.5, 5, water, sound , 242

図 C.6: 距離条件で効果音を再生する際のフォーマットと例

## 付録 D 本評価で用いた資料

図 D.1 から図 D.13 に評価で用いた資料を示す。

### 取り組んでいただく課題内容の説明

今回取り組んでいただく課題は、実際に行われている訓練とは異なりますので、ご了承お願い致します。説明資料をご覧になりながら、取り組んでください。

### シナリオ作成課題 1：漏水

想定状況：  
屋内で漏水が2か所発生した。

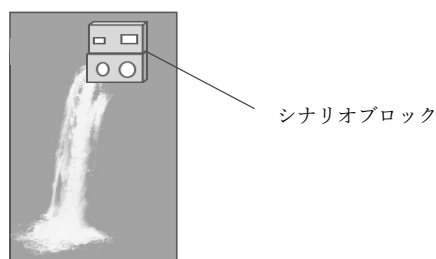
課題取り組み時間目安 : 15分

ARで表現する部分：

- 2か所の漏水の再生  
訓練開始から6秒後、20秒後に発生、  
場所は自由
- 2か所の漏水の停止  
訓練開始から25秒後、30秒後に停止、  
場所は漏水再生のシナリオブロックから1メートル以内



### タイムラインのイメージ図



### 漏水のイメージ図

図 D.1: シナリオ作成課題 1 の課題内容を記した資料

## シナリオ作成課題2： 放火時

### 想定状況：

部屋に侵入者が侵入して、部屋の紙類（図1）に放火する。火が燃え広がり、部屋内の爆発性物体（図2）に引火し、爆発する。訓練体験者は電話（図3）に近づくことで通報を行い警察が来るのを待つ。

課題取り組み時間目安： 30分



図1 紙類



図2 爆発性物体



図3 電話

### ARで表現する部分：

- ・ 侵入者が侵入する（場所、時間は自由）
- ・ 侵入者が火をつける様子（かがむ動作をする、場所は紙類の近く、時間は自由）
- ・ 火が発生する様子（場所は紙類の近く、時間は自由）
- ・ 火が燃え広がる様子（場所は紙類と爆発性物体の間、時間は自由）
- ・ 爆発が発生する様子（場所は爆発性物体の近く、時間は自由）
- ・ ユーザーが通報する様子（場所は電話の近く、1メートル以内に近づいたら）
- ・ 通報の一分後に警察が駆けつける様子（場所は自由、時間はユーザーが通報した1分）

### タイムラインのイメージ図

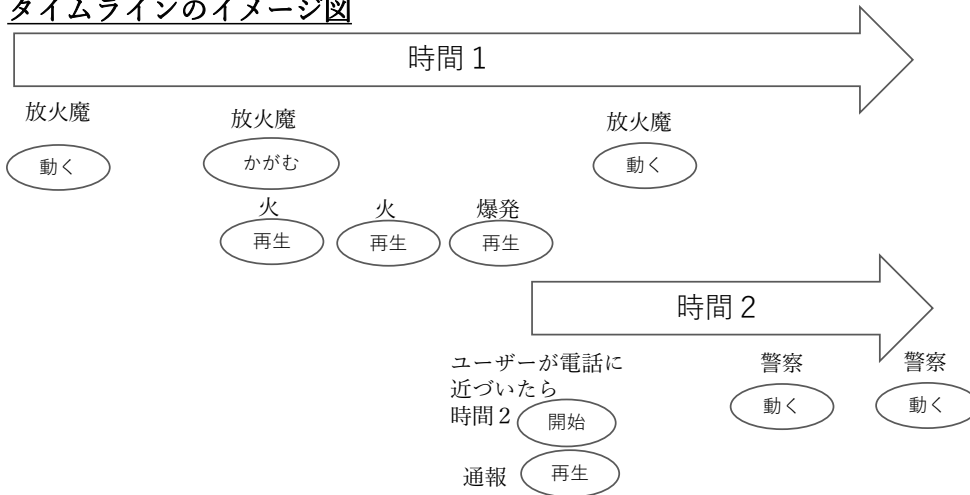


図 D.2: シナリオ作成課題2の課題内容を記した資料



## 自由作成課題

本システムにとらわれず、現場で必要とされていると思うAR訓練を考えていただき、下の記入欄に記入してください。  
その後、考えていただいたAR訓練を、本システムで作成できるかを試していただきます。

課題取り組み時間は30分程度です。わからないことがあれば、またAR訓練が思いつかないなどがあれば、ご自由にお聞きください。

**記入欄**（考えたAR訓練を記入してください）

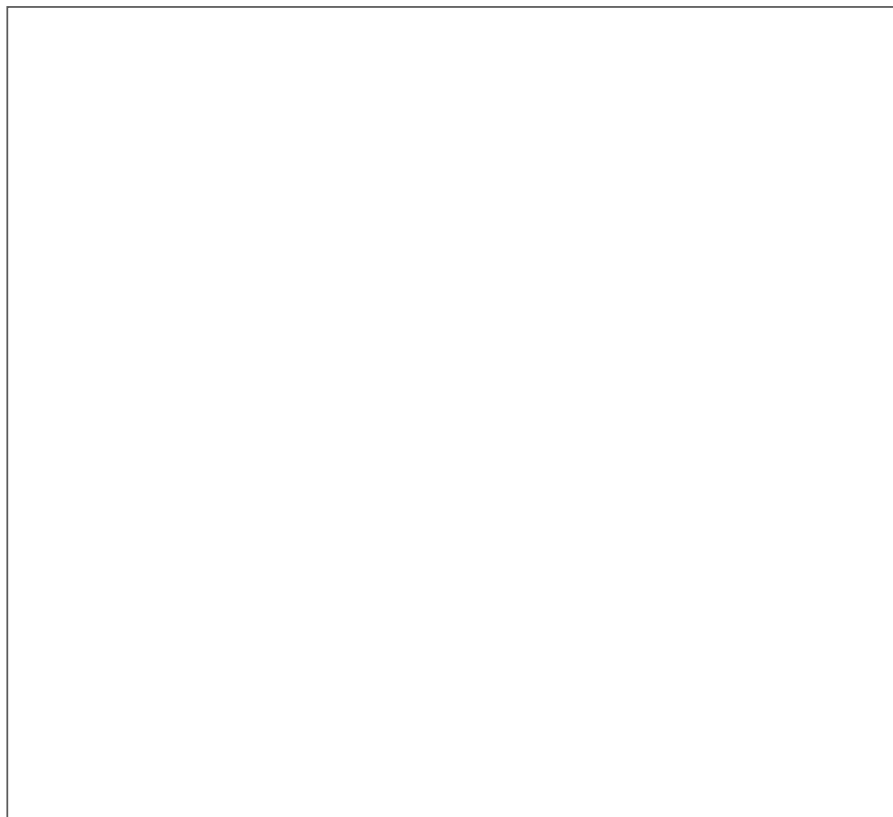


図 D.3: 自由作成課題の課題内容を記した資料

# 説明資料

京都大学大学院エネルギー科学研究科  
エネルギー社会・環境科学専攻  
エネルギー情報学研究室

この度は評価にご協力いただき、ありがとうございます。評価に先立ち、評価に関する説明および諸注意といくつかお願いがございますので、熟読の上、ご理解とご協力をお願いいたします。

## 評価の目的

AR訓練環境構築システム（以下ARTM）を用いて、AR訓練を作成できるかどうかを評価していただきます。

## 評価の概要

評価に要する時間は？時間程度です。AR訓練を作成する2つの課題に取り組んでいただきます。一つ目はこちらが示した訓練シナリオを作成していただく、「シナリオ作成課題」、二つ目は評価者自らが考えた訓練シナリオを作成していただく、「自由作成課題」です。それぞれについてアンケートに答えていただきます。

## 評価スケジュール

20分	評価の目的の説明、ARTMの説明（ビデオ）
15分	ARTMの練習
2分	課題内容の説明
15分	シナリオ作成課題 1
30分	シナリオ作成課題 2
5分	アンケート
30分	自由作成課題
5分	アンケート

1

図 D.4: 評価概要の事前説明に用いた資料



図 D.5: ARTM の説明ビデオのキャプチャ画像

# AR Training Maker : ARTM

ARTMを用いて可能なことは以下の二つです。

1. AR訓練環境の構築
2. AR訓練環境の体験

ARTMは以下の要素で構成されています。

1. プログラムブロック
2. カメラ付きタブレット

## **ARTMの使用方法**

### **1.AR訓練環境の構築について**

例として、以下のようなシナリオを作成する場合について、動画を用いながら説明します。

(例) 訓練開始から10秒後、電子レンジから発火する

#### **1-1 プログラムブロックを組み立てる**

プログラムブロックを組み立てることで、どのCGアニメーションをどの条件でどこに表示するかを決定できます。詳しいプログラムブロックの組み立て方については、2ページ以降で説明いたします。例を実現したい場合、時間、数字、火、再生、というパーツブロックを使用します。パーツブロックを、それらの組み合わせ方を定めるベースブロックに組み込むことで、一つのシナリオが完成します。その完成した組を、シナリオブロックと呼ぶことにします。

#### **1-2 組み立てたシナリオブロックを訓練環境中に設置する。**

1-1で組み立てたシナリオブロックを、CGアニメーションを表示させたい場所に設置してください。例では電子レンジに設置しています。

#### **1-3 シナリオブロックをカメラ付きタブレットを用いて撮影する**

カメラ付きタブレットを用いて、1-2で設置したすべてのシナリオブロックを撮影してください。認識されたシナリオブロックは緑色の枠でハイライトされます。

### **2. AR訓練環境の体験について**

カメラ付きタブレットに表示される映像を見ながら、訓練を遂行してください。タブレット上で、シナリオブロックの内容に応じたCGアニメーションを確認することができます。

2

図 D.6: 説明ビデオ内容の補足資料

## プログラムブロックの概要

プログラムブロックは、どのCGをどういう条件で、どこに表示するかを定められるブロックです。

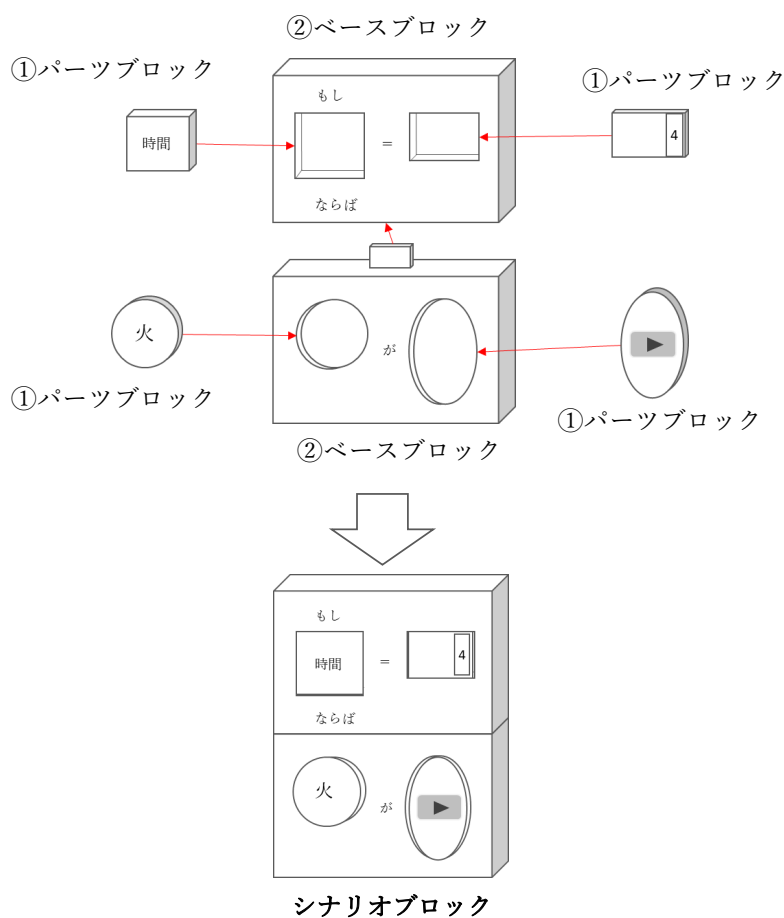
プログラムブロックは大きく分けて

①パーツブロック

②ベースブロック

の二種類があります。

パーツブロックは登場人物や登場物体（例「炎」、「時間」、「警察官」）を表し、ベースブロックはパーツブロックの関係を決めます。パーツブロックをベースブロックに組み込むことで、どのCGをどういう条件で、どこに表示するかを定められます。この組をシナリオブロックと呼ぶことにします。

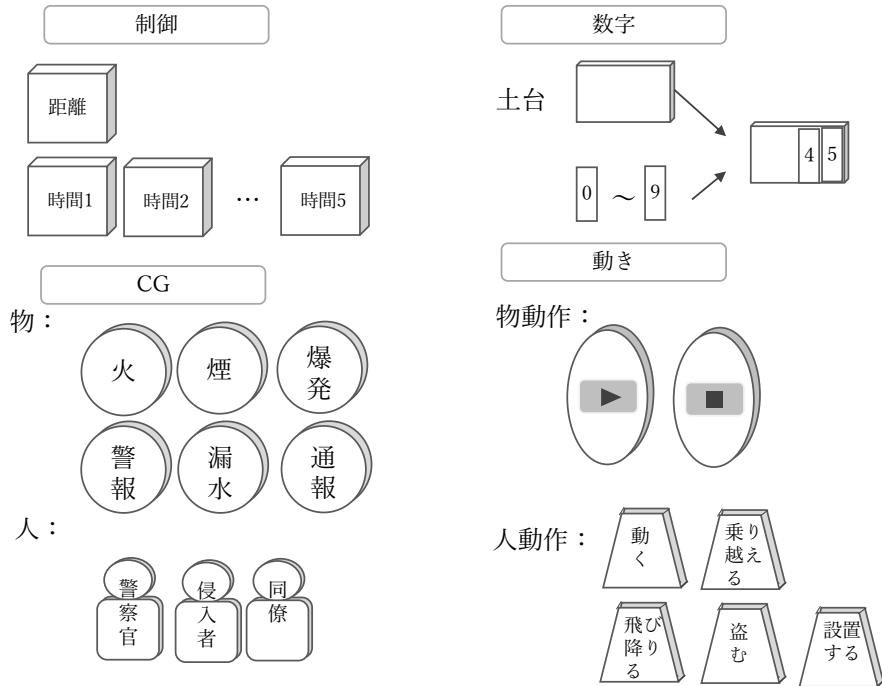


3

図 D.7: プログラムブロックの説明資料 1

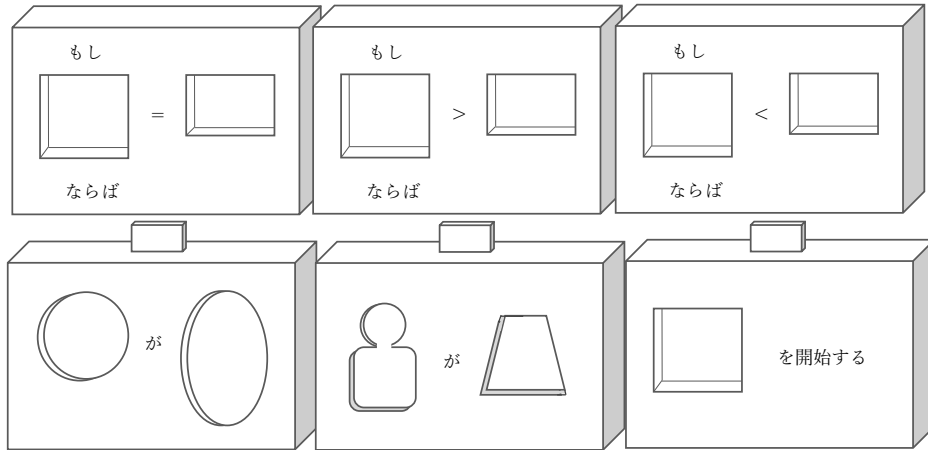
## プログラムブロックの種類

### ① パーツブロック



### ② ベースブロック

ベースブロックのくぼみの形ははめ込められるパーツブロックと同じ形をしています。別の形のパーツブロックをはめ込むことはできません。



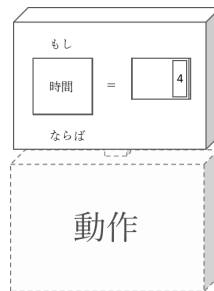
4

図 D.8: プログラムブロックの説明資料 2

## よくある間違い

以下の間違いがある場合、カメラ付きタブレットで認識することができません

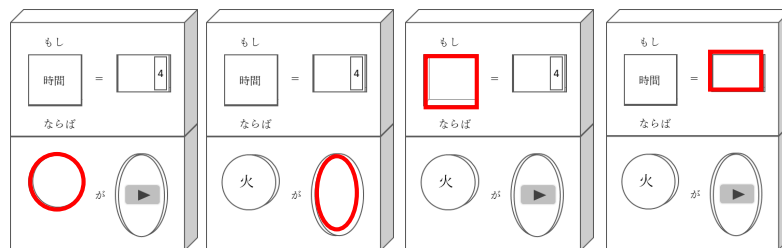
### ◆ 条件しか設定されておらず、結果がない



### ◆ 結果しか設定されておらず、条件がない



### ◆ パーツブロックがはめ込まれていない場所がある



12

図 D.9: プログラムブロックの説明資料 3

## 注意点

撮影時は、撮影された画像により自己位置の推定（トラッキング）を行っています。画像がブレない程度に緩やかに動いてください。急に振り返るなど素早い動きをすると、トラッキングに失敗してしまいます。トラッキングに失敗した時は、画面上に赤い文字が表示されるので（図）、トラッキングに失敗する以前の場所へと戻ってください。

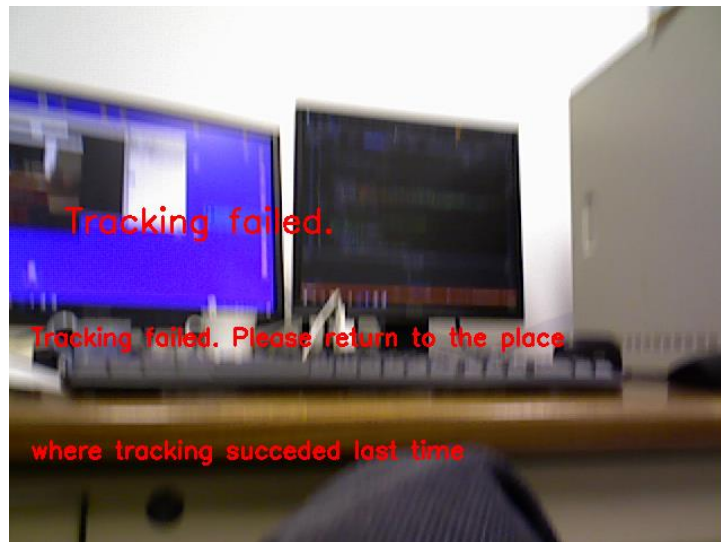


図 トラッキング失敗時



## この評価に関する問い合わせ

評価内容や結果について、ご不明な点や詳しく知りたいことがございましたら、下記に連絡してください。

- ・お問い合わせ  
京都大学大学院エネルギー科学研究科  
エネルギー社会・環境科学専攻 エネルギー情報学研究室  
住所：〒606-8501 京都市左京区吉田本町  
電話：075-753-5613  
Mail（不利益の発生等に関して）：hirotake@energy.kyoto-u.ac.jp

図 D.11: プログラムブロックの説明資料 5

## アンケート用紙（シナリオ作成課題）

システムに関する以下の項目について、次の5段階評価のうち最も当てはまると思うものを、○をつけてお答えください。

1. そう思わない 2. あまりそう思わない 3. どちらでもない 4. ややそう思う 5. そう思う

	そう思わない			そう思う	
	1	2	3	4	5
① どのパーツブロックがどの機能を表すか理解できた。					
② ベースブロックにどのパーツブロックをはめ込められるか理解できた。					
③ 時間1を条件として設定することは容易だった。					
④ 距離を条件として設定することは容易だった。					
⑤ 時間1と時間2の違いを理解することは容易だった。					
⑥ 物CG（火や漏水など）の再生を設定することは容易だった。					
⑦ 物CG（火や漏水など）の停止を設定することは容易だった。					
⑧ 人物CG（警察官、侵入者など）の動きを設定することは容易だった。					
⑨ CGの表示位置を設定することは容易だった。					
⑩ CGの表示向きを設定することは容易だった。 （漏水の流れや、人の立つ向きなど）					
⑪ シナリオブロックを撮影することは容易だった。					
⑫ 想像していた通りにCGが表示された。					
⑬ 初めて使う人でも簡単にシステムを使用できる。					

図 D.12: ARTM の容易さに関するアンケート用紙

## アンケート用紙(自由作成課題)

1. そう思わない 2. あまりそう思わない 3. どちらでもない 4. ややそう思う 5. そう思う

- |  |   |   |   |   |   |
|--|---|---|---|---|---|
| ① 現場で必要とされている訓練を作成するのに必要な物 CG (火、漏水など) を備えている。                 | 1 | 2 | 3 | 4 | 5 |
| ② 現場で必要とされている訓練を作成するのに必要な人 CG (警察官など) を備えている。                  | 1 | 2 | 3 | 4 | 5 |
| ③ 現場で必要とされている訓練を作成するのに必要な物 CG の動き (再生、停止) を備えている。              | 1 | 2 | 3 | 4 | 5 |
| ④ 現場で必要とされている訓練を作成するのに必要な人 CG の動き (動く、乗り越える、飛び降りる、かがむ) を備えている。 | 1 | 2 | 3 | 4 | 5 |
| ⑤ 現場で必要とされている訓練を作成するのに必要な条件 (時間、ユーザーとの距離など) を備えている。            | 1 | 2 | 3 | 4 | 5 |

システム全体で、良かった点や改良すべき点、気づいたことなどあればご自由に記述してください。

評価者 (氏名 : \_\_\_\_\_)

図 D.13: 訓練シナリオの自由度に関するアンケート用紙