

ゲームエンジンを用いたレスキューロボットコンテストのための 画面表示システムの実装

Implementation of Screen Display System for Rescue Robot Contest Using Game Engine

栗栖 輝^{*1}, 小島 篤博^{*2}

Hikaru KURISU^{*1}, Atsuhiko KOJIMA^{*2}

^{*1}大阪府立大学現代システム科学域

^{*1}College of Sustainable System Sciences, Osaka Prefecture University

^{*2}大阪府立大学

^{*2} Osaka Prefecture University

Email: sza01109@edu.osakafu-u.ac.jp

あらまし: レスキューロボットコンテストで運用されている競技運営支援システムは、複数のサブシステムから構成されている。そのサブシステムの1つに、競技チームや観客に対して競技状況などを提示する画面表示システムがある。本研究では、観客に向けた視覚的な演出を考慮して、ゲームエンジン Unity を用いて画面表示システムを実装した。これにより、汎用性が向上するとともに、視覚的効果の豊かな表現が可能になった。

キーワード: レスキューロボットコンテスト, ゲームエンジン, Unity, 競技運営, システム開発

1. はじめに

都市部の大規模災害におけるレスキュー活動をテーマとしたレスキューロボットコンテスト（以下「レスコン」）⁽¹⁾が毎年開催されている。レスコンでは、競技を円滑に運営するための独自の競技運営支援システム（以下「レスコンシステム」）⁽²⁾が運用されているが、このシステムは、複数のサブシステムから構成されている。そのサブシステムの1つに、競技状況などを競技チームや観客に対して提示する画面表示システムがある。本研究では、ゲームエンジンを用いて、観客に向けた視覚的な演出を考慮した画面表示システムを実装する。

2. レスキューロボットコンテストについて

2.1 競技の概要

レスコンは、レスキュー活動の啓発や創造性の教育を目的として毎年開催されているロボットコンテストである。競技内容は、被災地を模擬したフィールド上で要救助者に見立てたレスキューダミー（以下「ダミー」）を遠隔操作型のロボットで救助するというものである。ダミーには、救助活動の際に受ける痛みを検知するセンサが内蔵されており、そのセンサが検知した数値や救助の達成度などが各チームの得点に換算される。

2.2 競技の流れ

競技は、チーム紹介、プレゼンテーション、作戦会議、レスキュー活動、活動報告および得点表示という流れで行われる。

2.3 レスコンシステム

レスコンシステムは、データベースを中心として、複数のサブシステムから構成されている。サブシス

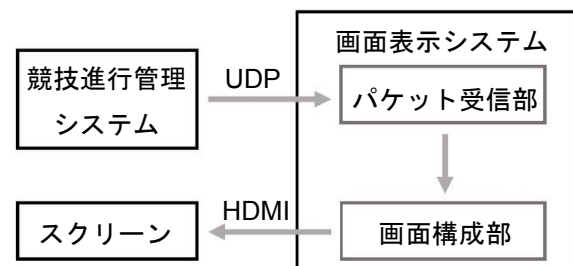


図1 画面表示システムの構成

テムの1つである画面表示システムは、従来 Visual Basic で実装されていたが、Windows でしか利用できないなどの問題があった。本研究では、ゲームエンジン Unity を用いて、様々なプラットフォームに対応するとともに、観客に向けた視覚的な演出を考慮した画面表示システムを実装する。

3. 画面表示システムの実装

3.1 システムの概要

画面表示システムは、競技状況などを競技チームや観客に対して提示するシステムである。このシステムの構成を図1に示す。図1が示すように、画面表示システムは、以下の2つの部分から構成されている。

- **パケット受信部**
別のサブシステムである競技進行管理システムから画面表示を指示する UDP パケットを受信する。
- **画面構成部**
受信した UDP パケットの情報を基に画面を構成する。

競技進行管理システムから受信する UDP パケットには、チーム名や得点、時間などの情報が含まれている。また、画面構成部で構成した画面は、HDMI 経由でスクリーンに投影される。

3.2 開発環境

開発環境として、ゲームエンジン Unity を採用した。Unity を採用した理由は以下の 3 点である。

- GUI ベースによる直観的な操作で画面を作成できる。
- ゲーム開発に用いられる様々な技術により、観客に向けた視覚的な演出が実現できる。
- Windows, Mac, iOS, Android など、様々なプラットフォームに対応している。

3.3 画面構成

画面表示システムは、以下の 6 種類の画面から構成されている。競技の進行に沿って、これらの画面が切り替えられる。

- **Main Window**
画面表示システムを起動した際に、最初に表示される画面である。スタートボタンを押すと、パケットの受信を開始する。
- **Stay Window**
競技開始前や競技進行の待機時間などに表示される画面である。
- **Intro Window**
チーム紹介の際に表示される画面である。チームの集合写真が表示される。
- **Presen Window**
プレゼンテーションと作戦会議の際に表示される画面である。残り時間の表示を行う。
- **DummyIndicator Window**
レスキュー活動の際に表示される画面である。ミッションの達成状況など、競技中の状況がリアルタイムに表示される。
- **Score Window**
レスキュー活動終了後に行われる得点表示の際に表示される画面である。チームの総合得点などが表示される。

3.4 非同期処理

画面表示システムでは、受信した UDP パケットの情報をリアルタイムに画面に反映させる必要がある。そのため、非同期処理によって、UDP パケットの受信と画面の構成を切り離して行っている。画面表示システムで行われている非同期処理の様子を図 2 に示す。別スレッド内で Unity API を使う場面があるが、Unity には、Unity API をメインスレッド外で操作できないという制限がある。そのため、別スレッド内で Unity API を使う際には、一時的にメインスレッドに処理を渡し、別スレッドはメインスレッドの処理を待機するという動作をする。

3.5 新しく追加した要素

DummyIndicator Window を図 3 に示す。Unity の機

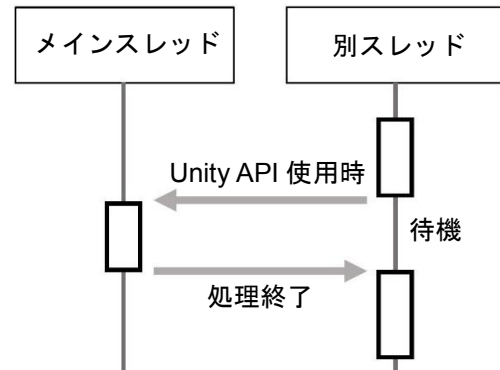


図 2 スレッド間の非同期処理

Bloom による発光でインパクトのある表現を実現



Particle を適用し、バーの減少を表現

図 3 DummyIndicator Window

能を利用して、DummyIndicator Window に以下の 2 つの要素を追加した。これにより、視覚的効果の豊かな表現が実現した。

- **Bloom**
光源から光が溢れるような演出ができる。
- **Particle**
粒子を発生させ、揮発するような効果を演出できる。

4. まとめ

本研究では、ゲームエンジン Unity を用いて画面表示システムを実装した。これにより、システムの汎用性が向上するとともに、視覚的効果の豊かな表現が可能になった。今回は、Unity の Bloom と Particle の機能を利用したが、Unity では、3D での画面表示など、他にも様々な機能が利用できる。今後の課題は、Unity で利用できる様々な機能を活かし、より効果的な表現を実現することが挙げられる。

参考文献

- (1) レスキューロボットコンテスト公式サイト：
<https://www.rescue-robot-contest.org/>
- (2) 小島篤博, 小枝正直, 山内仁: “レスキューロボットコンテストのための競技運営支援システムの開発と評価”, 電子情報通信学会論文誌 D, Vol.J93-D, No.10, pp.2317-2325 (2010)