

react cop2

ono

2023 年 12 月 17 日

目次

1	はじめに	1
1.1	研究背景	1
1.2	研究課題	1
1.3	研究目的	1
1.4	本論文の構成	1
2	関連研究	1
2.1	COP の概要	1
2.2	React COP	1
2.3	EventCJ に複合層を導入	1
3	提案手法	1
3.1	改善点	1
(1)	改善点一覧	1
(2)	layer の de/active 時に新しいレイヤーを定義できないようにする	2
(3)	layer params は in/active の両方の状態を持つ	2
(4)	layer params は layer の in/active 状態に依存をするようにしたい	2
(5)	layer params の値を入れるときに新しい layer を定義できないようにする	2
(6)	layer grop 的なのをつか	2
(7)	layer の活性化条件を定義できる	2
(8)	layer の活性化は排反	2
3.2	実装する内容	2
(1)	typescript での実装	2
(2)	テストの追加	2
3.3	評価方法	2
4	実装	2
4.1	ts の導入	2
4.2	テストの導入	2
4.3	各種、具体的な実装内容	2
5	評価	2
5.1	できるようになったこと	3
6	まとめ	3
7	参考文献	3

1 はじめに

1.1 研究背景

1.2 研究課題

1.3 研究目的

1.4 本論文の構成

2 関連研究

2.1 COP の概要

2.2 React COP

2.3 EventCJ に複合層を導入

3 提案手法

本研究では、関連研究にある複合層、多層の機能追加 react cop の改善点を洗い出し、それを解決するための機能を追加した。本章では、本研究で提案する手法について述べる。

3.1 改善点

(1) 改善点一覧

- layer の de/active 時に新しいレイヤーを定義できないようにする
- layer params は in/active の両方の状態を持つ
- layer params は layer の in/active 状態に依存をするようにしたい
- layer params の値を入れるときに新しい layer を定義できないようにする
- layer grop 的なのをつか
- layer の活性化条件を定義できる
 - － 複合層
 - － 多層
- layer の活性化は排反
- typescript での実装
- テストの追加

- (2) layer の de/active 時に新しいレイヤーを定義できないようにする
- (3) layer params は in/active の両方の状態を持つ
- (4) layer params は layer の in/active 状態に依存をするようにしたい
- (5) layer params の値を入れるときに新しい layer を定義できないようにする
- (6) layer grop 的なのをつか
- (7) layer の活性化条件を定義できる
- (8) layer の活性化は排反

ソースコード 1: hoge

```
1      // このとき、Float と Integer のレイヤの活性化は排反でよい気がする
2      const [getHoge, setHoge] = useLayerParams('', ["Float", "Integer"]);
3
4      // いちいち切り替えがめんどろ
5      layerManager.deactivateLayer("Integer");
6      layerManager.activateLayer("Float");
7
8
9      // 排反ではないから 2 つ条件含むのどうなん?
10     <Layer condition={layerState.Float && !layerState.Integer}>
```

3.2 実装する内容

- (1) typescript での実装
- (2) テストの追加

3.3 評価方法

- 実装前と後で、できることの違いを比較する。

4 実装

- 4.1 ts の導入
- 4.2 テストの導入
- 4.3 各種、具体的な実装内容

5 評価

本章では、提案手法の評価を行う。

5.1 できるようになったこと

6 まとめ

7 参考文献