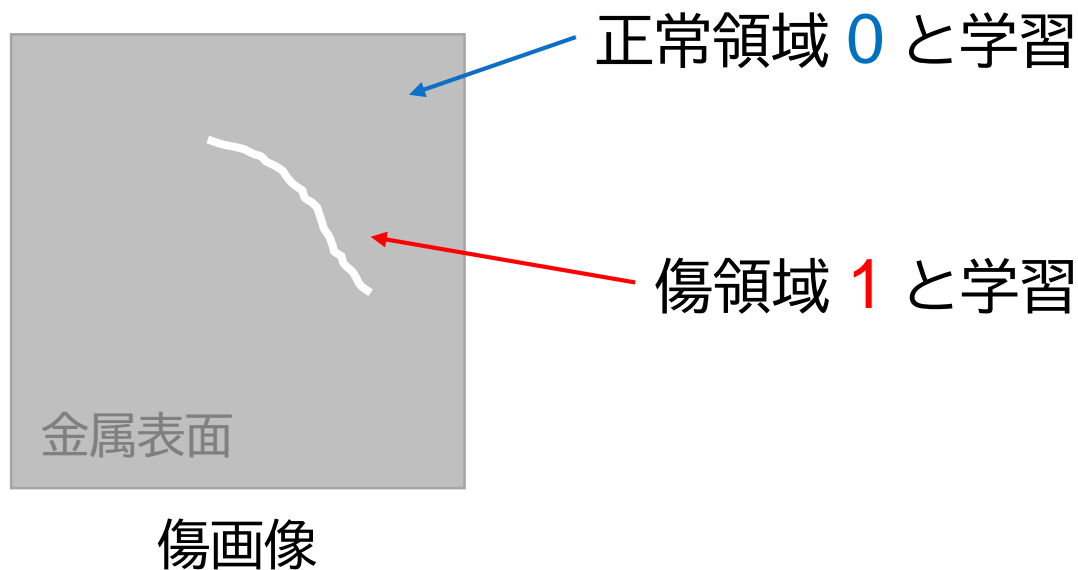


# CNN2値分類モデルの利用

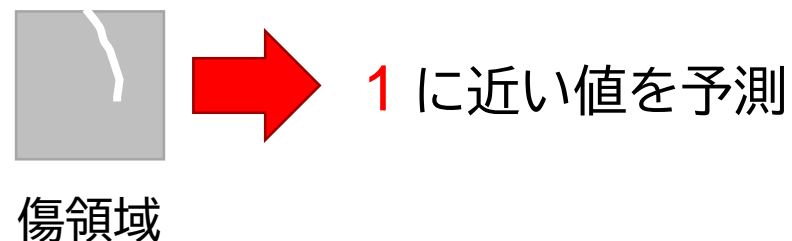
Kentaro Fujita

# 欠陥検出手法

- CNN2値分類モデルの利用



----- 学習していない画像の場合 -----



# CNN2値分類モデルの実装

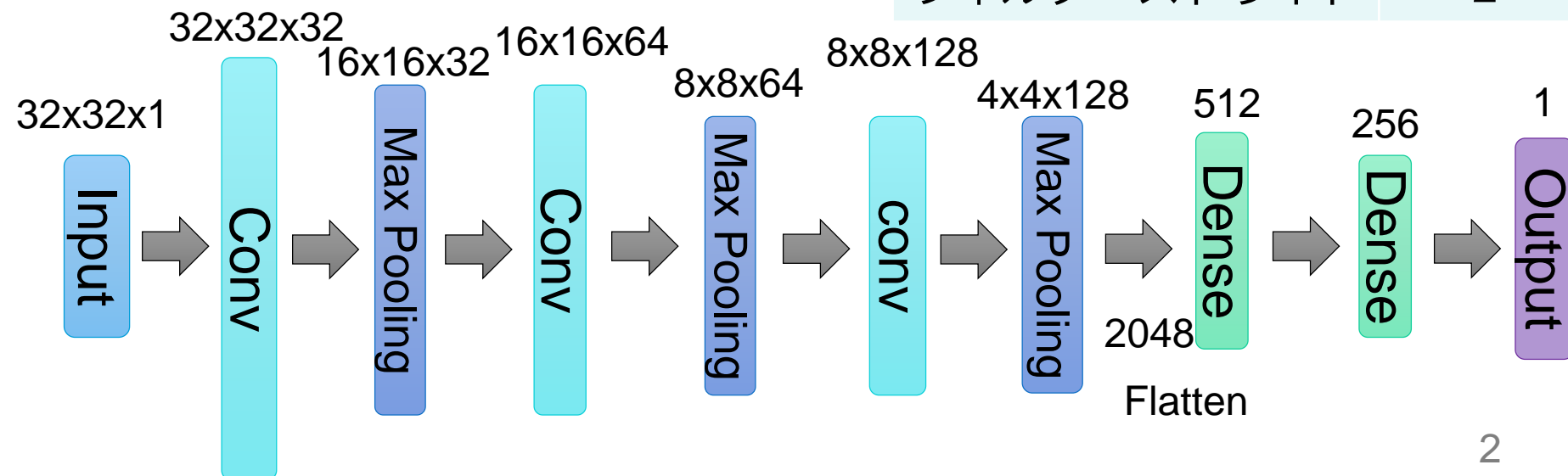
- 損失関数

Binary Cross Entropy

$$L = -t \log y - (1 - t) \log(1 - y)$$

- ネットワーク構造

項目	値
ブロックサイズ	32 × 32
学習エポック数	10
バッチサイズ	32
フィルター数	32
フィルターサイズ	2 × 2
フィルターストライド	2



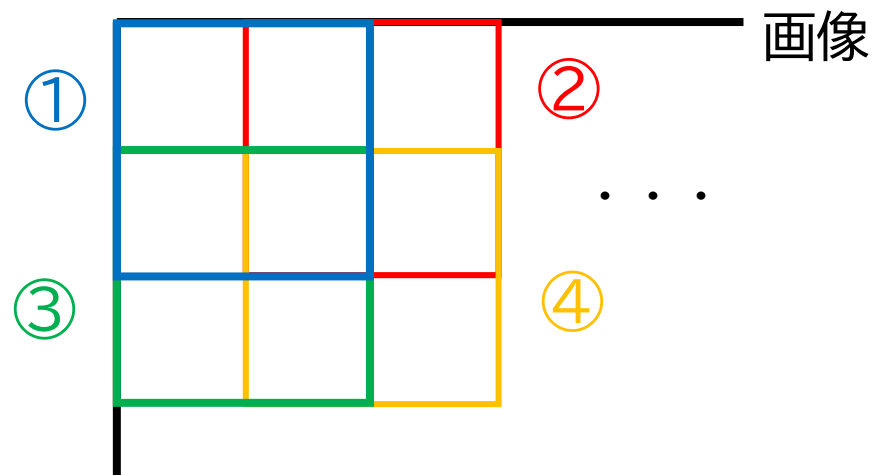
# 学習・検査手法

- 学習

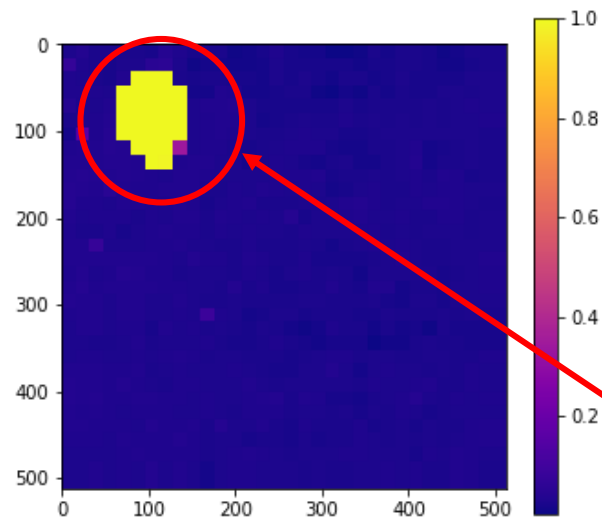
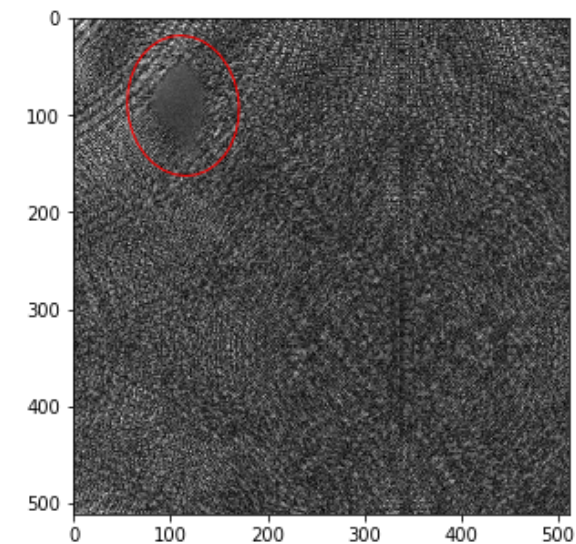
学習画像から **ランダム** 位置のブロックを切り出し学習  
⇒ 学習データの分散を変えず，サンプル数を増やすため

- 検査

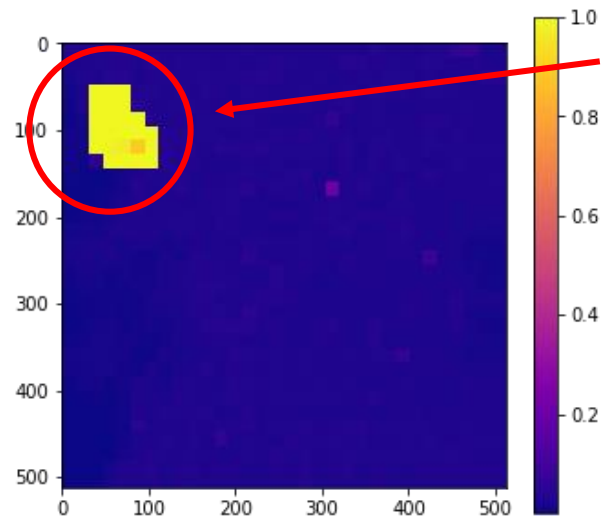
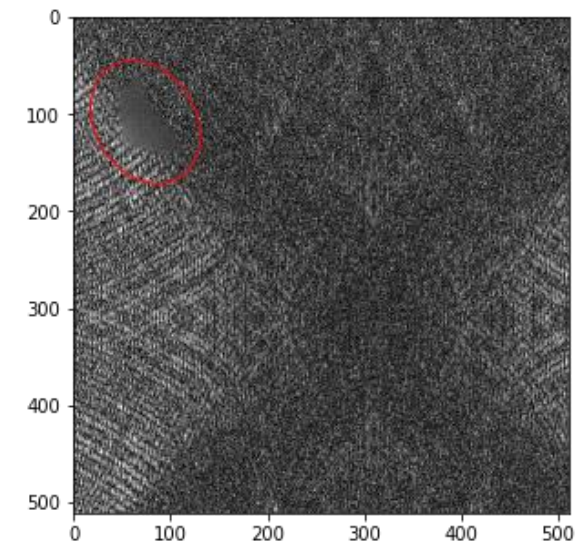
検査画像を **格子状** に切り出し，半ブロックずらしつつ検査  
⇒ 欠陥領域の見逃しを防ぐため



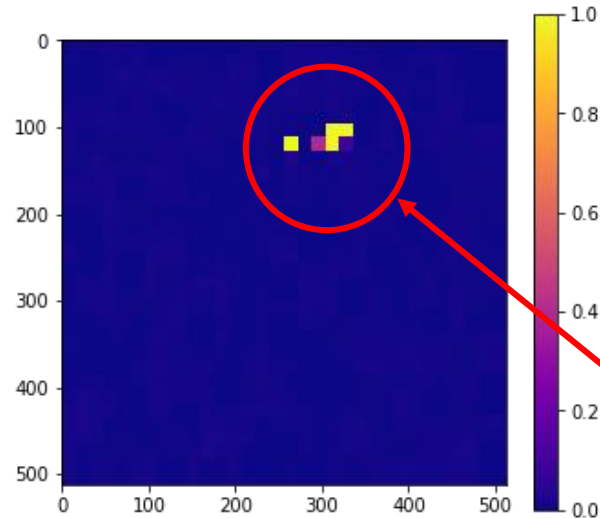
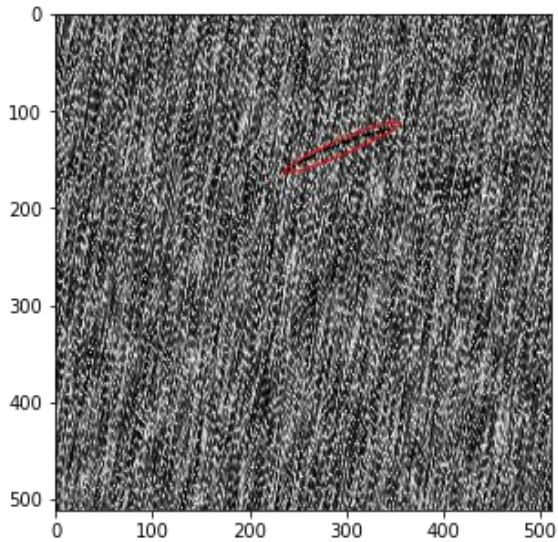
# 実行結果 Class1



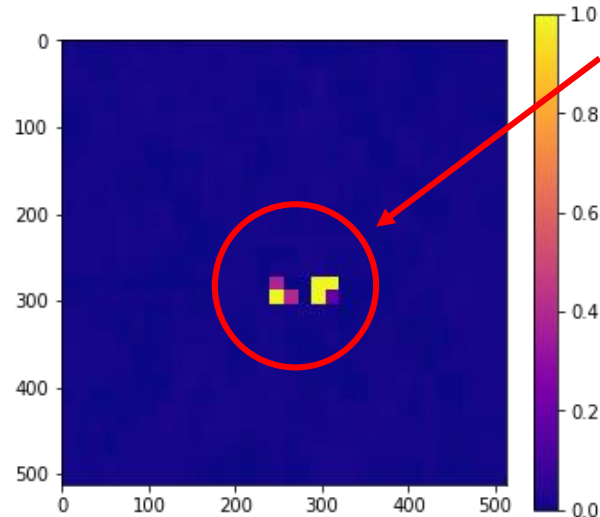
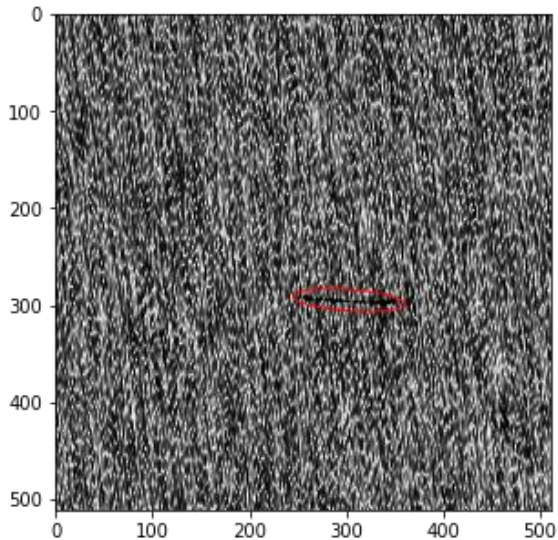
値が1 近くなっている  
⇒ 欠陥の検出が可能である



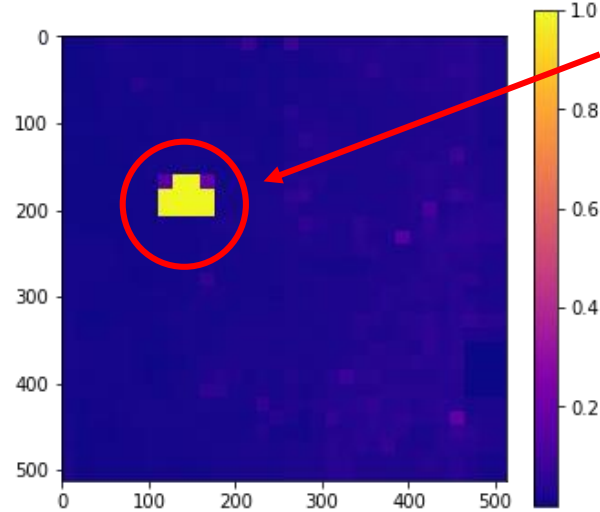
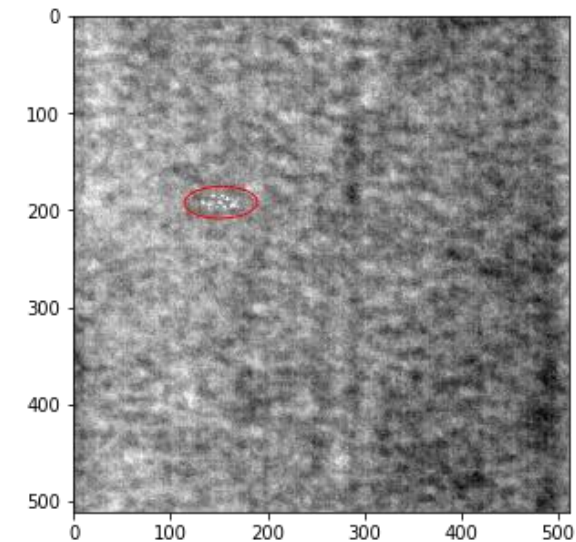
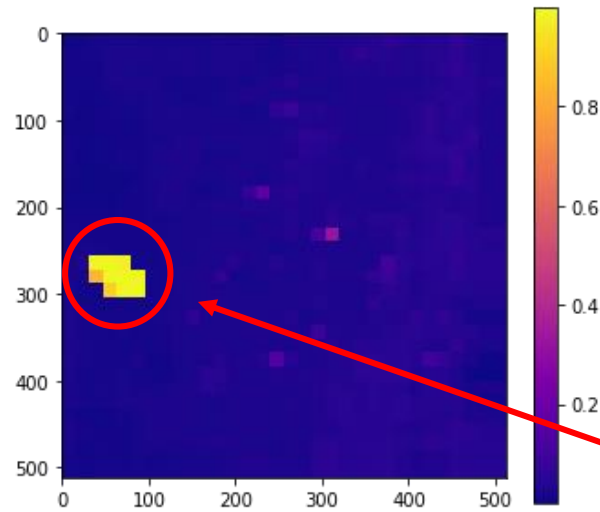
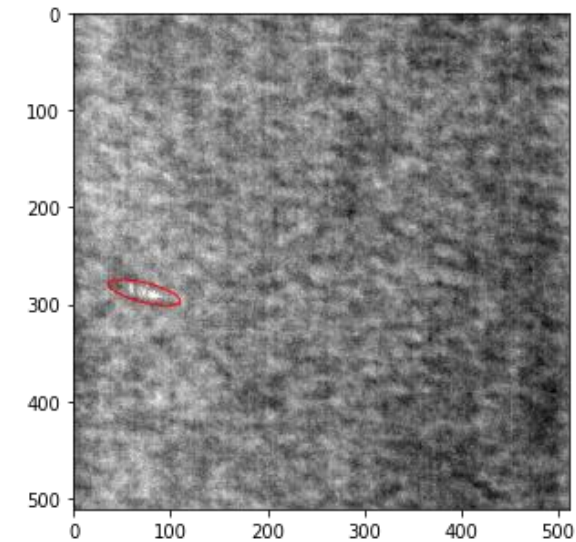
# 実行結果 Class2



値が1 近くなっている  
⇒ 欠陥の検出が可能である



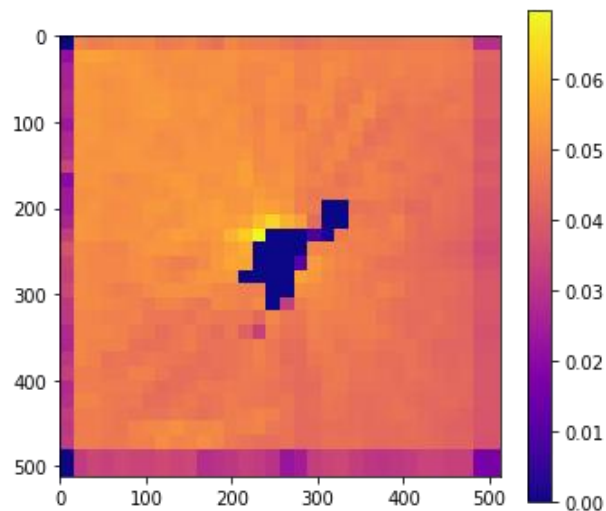
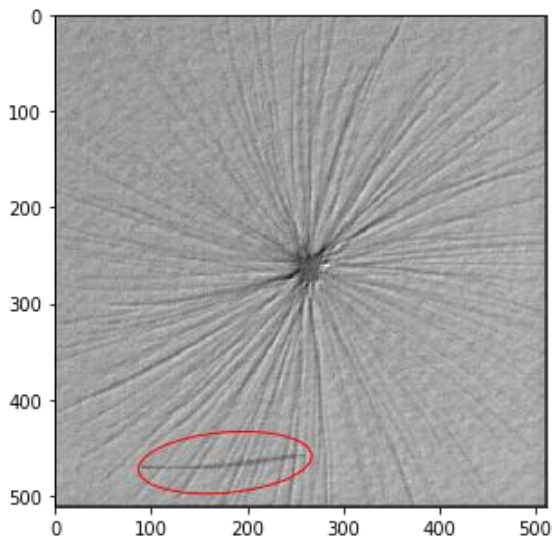
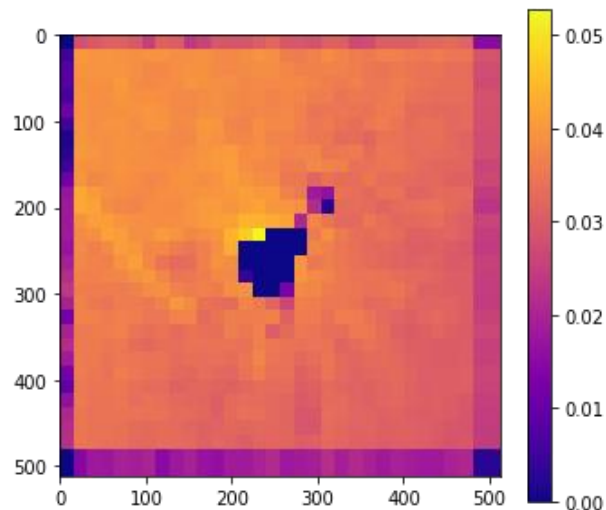
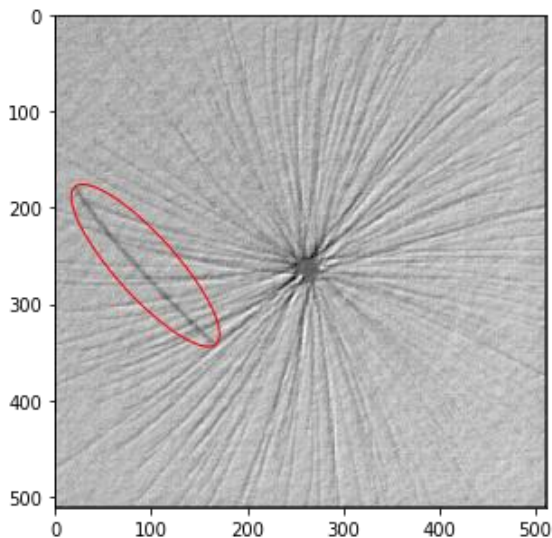
# 実行結果 Class3



値が1 近くなっている  
⇒ 欠陥の検出が可能である



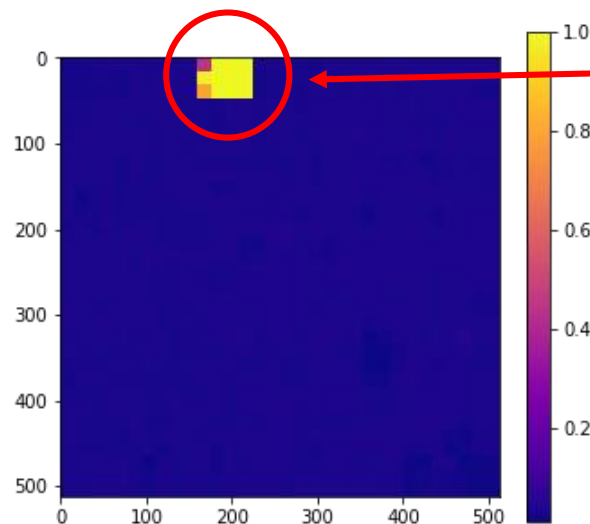
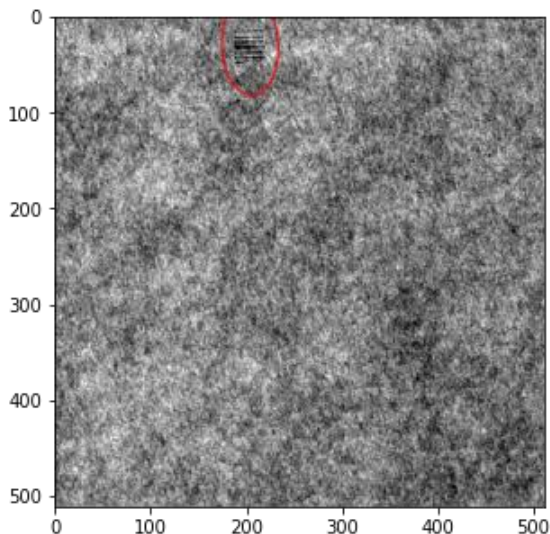
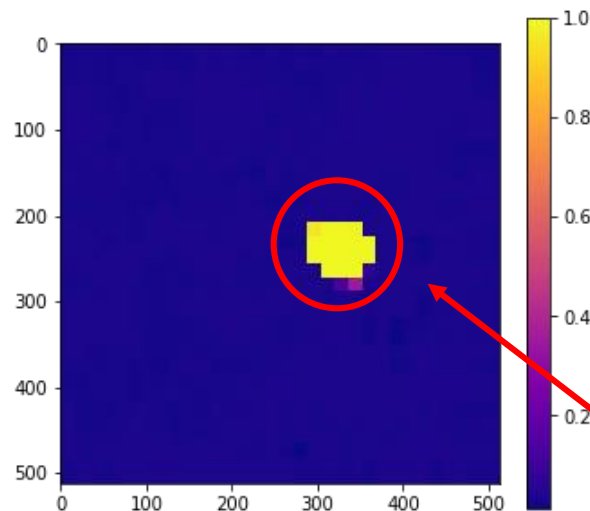
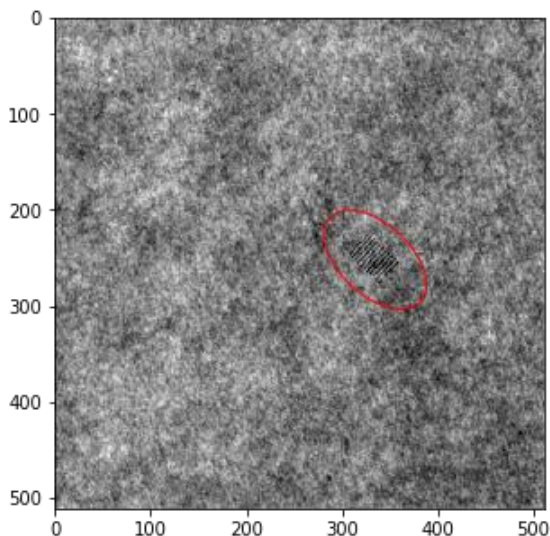
# 実行結果 Class4



全体的に値が小さい  
⇒ 傷を正しく  
学習できていない

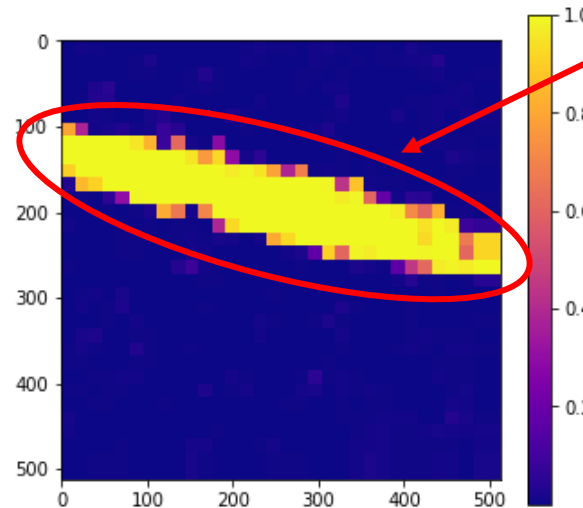
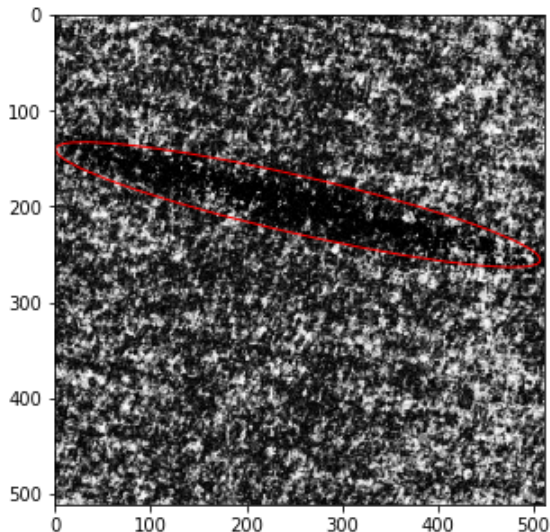
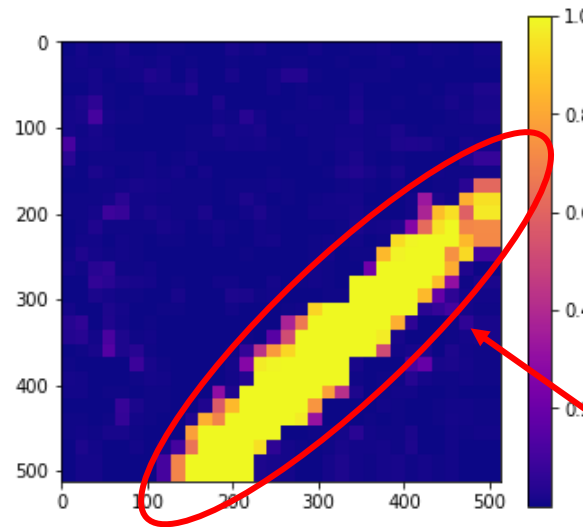
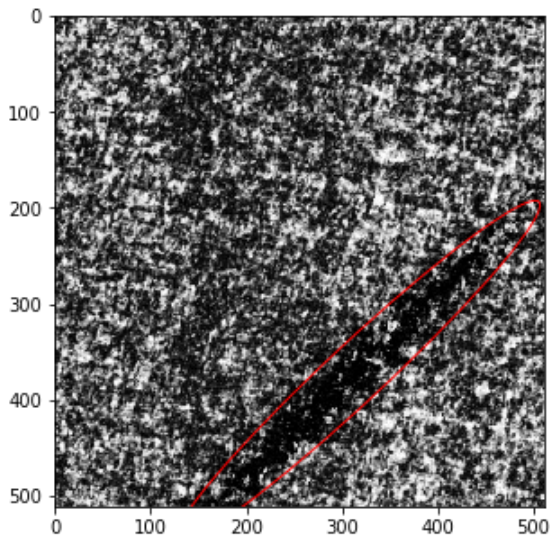


# 実行結果 Class5



値が1 近くなっている  
⇒ 欠陥の検出が可能である

# 実行結果 Class6



値が1 近くなっている  
⇒ 欠陥の検出が可能である

# まとめ

## 1. Class 4 以外

- 現状，十分傷が検出可能である。  
⇒ 前処理が必要ない

## 2. Class 4

- いずれかの前処理を施す必要がある。

## • コメント

傷画像が十分に用意できる場合，この手法は有用であるといえる。  
一方で，傷サンプルが豊富な例は多くない。