

画像処理によるフィラメント 数のカウント方法の開発

繊維・高分子科	研究員	小林慶祐
電子・情報科	研究員	三瓶史花
ロボット・制御科	研究員	根本大輝

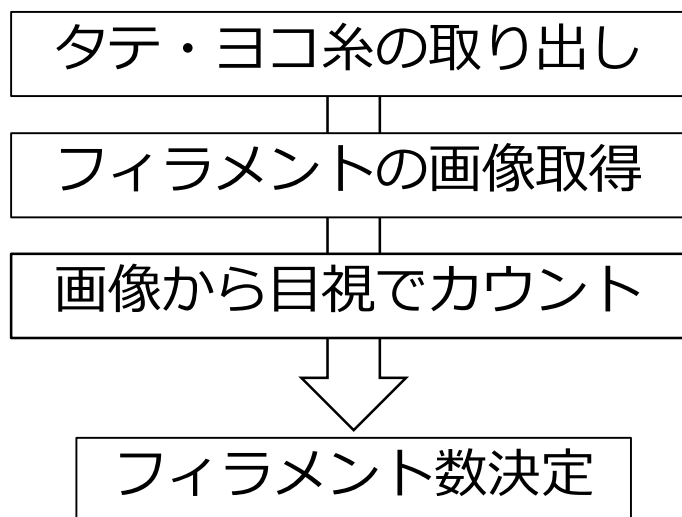
質問はメールにて事務局までお気軽にお問い合わせください。
問い合わせ先：福島県ハイテクプラザ 企画連携部産学連携科
e-mail : hightech-renkei@pref.fukushima.lg.jp

研究背景

○新たな生地を開発するときに、比較となる生地詳細を明らかにする必要とあり、目付や密度などに加えて糸のフィラメント(単繊維)数のカウントを行うことがある。

Ex.産業資材(太い、フィラメント数多)、高機能性フィラメントの詳細(異形等)

従来の測定方法



従来の取得画像例

数え間違い防止のため複数回実施

従来の手法の課題

- ・本数が多い場合に数え間違い防止のため複数回実施することで時間がかかる
- ・脱落による誤差が発生する

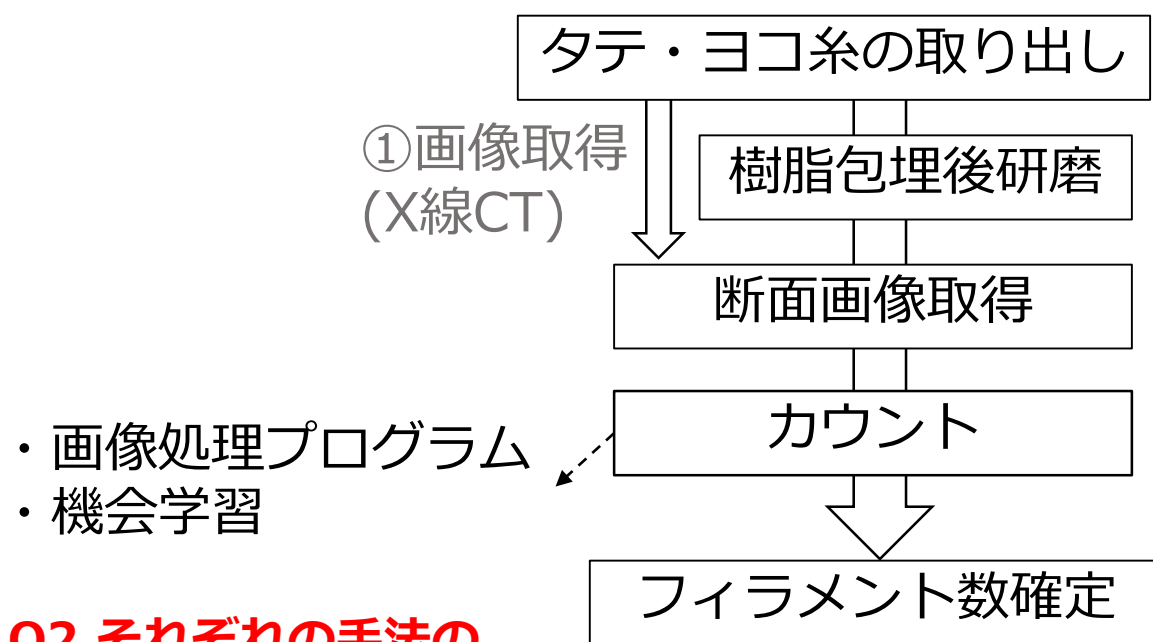
フィラメント数を決めるために必要な、**作業時間を30%低減**し、かつ誤差が少ない方法の開発を目的として研究を行った。

検証方法

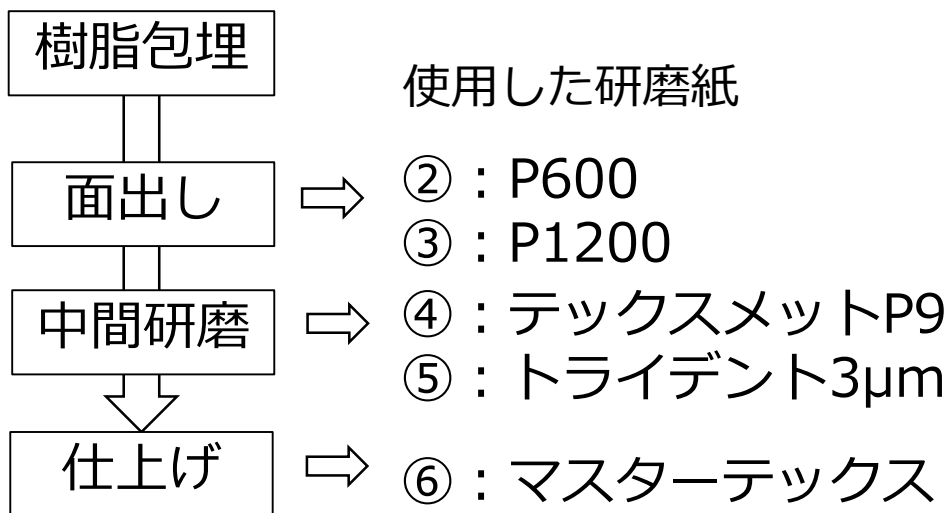
○作業時間を低減しつつ、誤差を少なくする方法として糸を小分けすることなく、そのままの状態での測定が可能な方法を検討した。

○対象とした糸：断面が円形の合成繊維(直径約20 μ m)

検討した測定方法



樹脂包埋後、研磨工程詳細



Q2.それぞれの手法の
カウントの精度は？

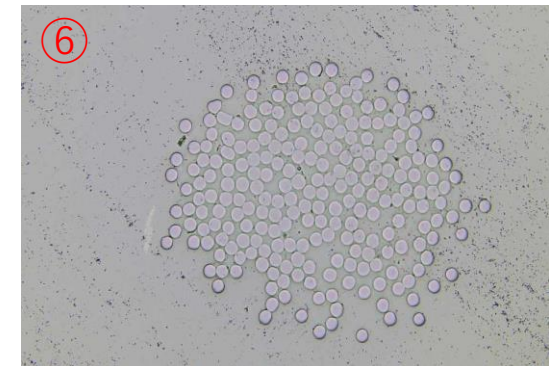
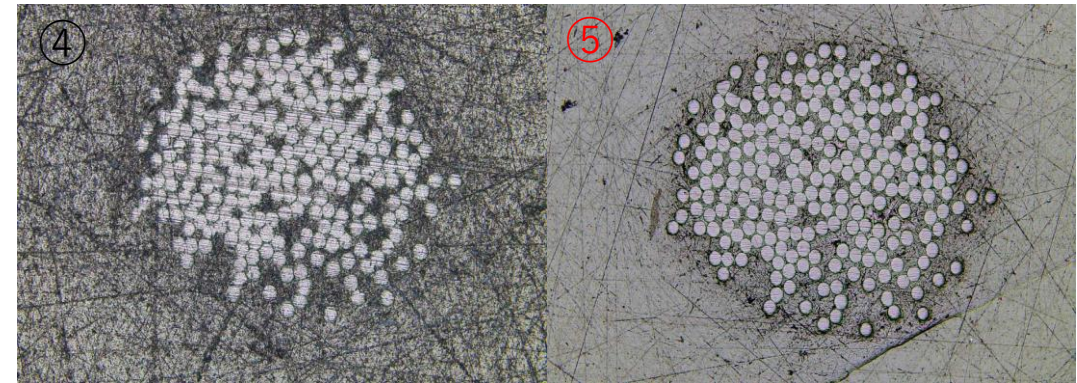
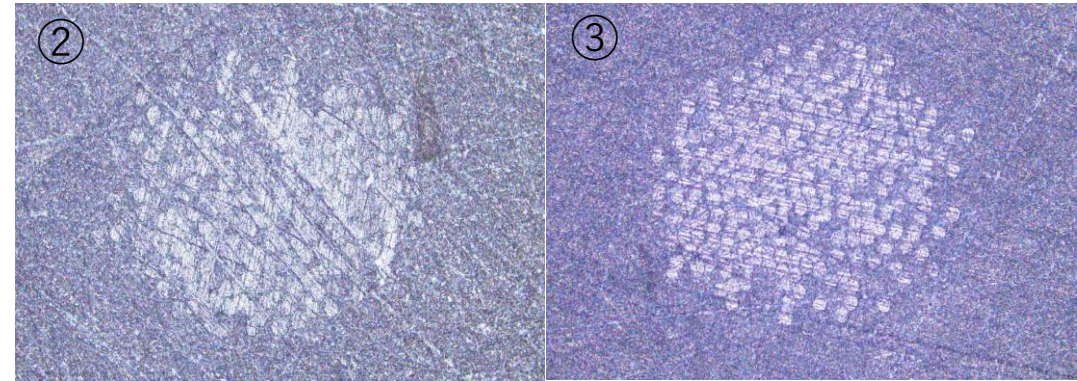
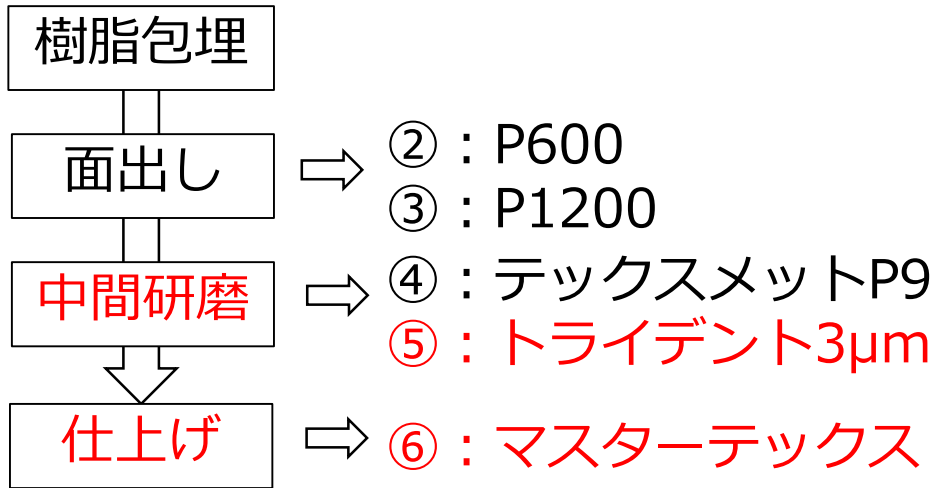
Q3.従来と比較して作業時間は？

Q1.カウントのためにどこまで研磨が必要か？

樹脂包埋・研磨後の断面画像の観察

○どこまで研磨が必要か確認するために、各研磨後に断面を観察した。

樹脂包埋後、研磨工程詳細



④中間研磨でもカウント可能、安定した画像を得ることが難しい
⑤中間研磨、⑥仕上げ研磨に着目して、画像処理及び機械学習で
カウントすることができるか検証した。

画像処理によるフィラメント数のカウント

- Python3を用いて画像内の円を検出するプログラムを作成しフィラメントを検出した。
- 結果は検出率で評価した。

$$\text{検出率(\%)} = \frac{\text{(画像内のフィラメント数)}}{\text{(検出したフィラメント数)}} \times 100$$

```
kernel_gauss_circle - XE帳
ファイル 編集 表示

import cv2
import numpy as np

count = 0

def make_sharp_kernel(k: int):
    return np.array([
        [-k / 9, -k / 9, -k / 9],
        [-k / 9, 1 + 8 * k / 9, -k / 9],
        [-k / 9, -k / 9, -k / 9]
    ], np.float32)

img = cv2.imread("test1.jpg")
kernel = make_sharp_kernel(1)
img = cv2.filter2D(img, -1, kernel).astype("uint8")

img_gauss = cv2.GaussianBlur(img, (3, 3), 3)
imgs = cv2.hconcat([img, img_gauss])

gray = cv2.cvtColor(img_gauss, cv2.COLOR_BGR2GRAY)
gray = cv2.medianBlur(gray, 5)

circles = cv2.HoughCircles(gray, cv2.HOUGH_GRADIENT, dp=1, minDist=19,
                           param1=100, param2=18, minRadius=10, maxRadius=20)

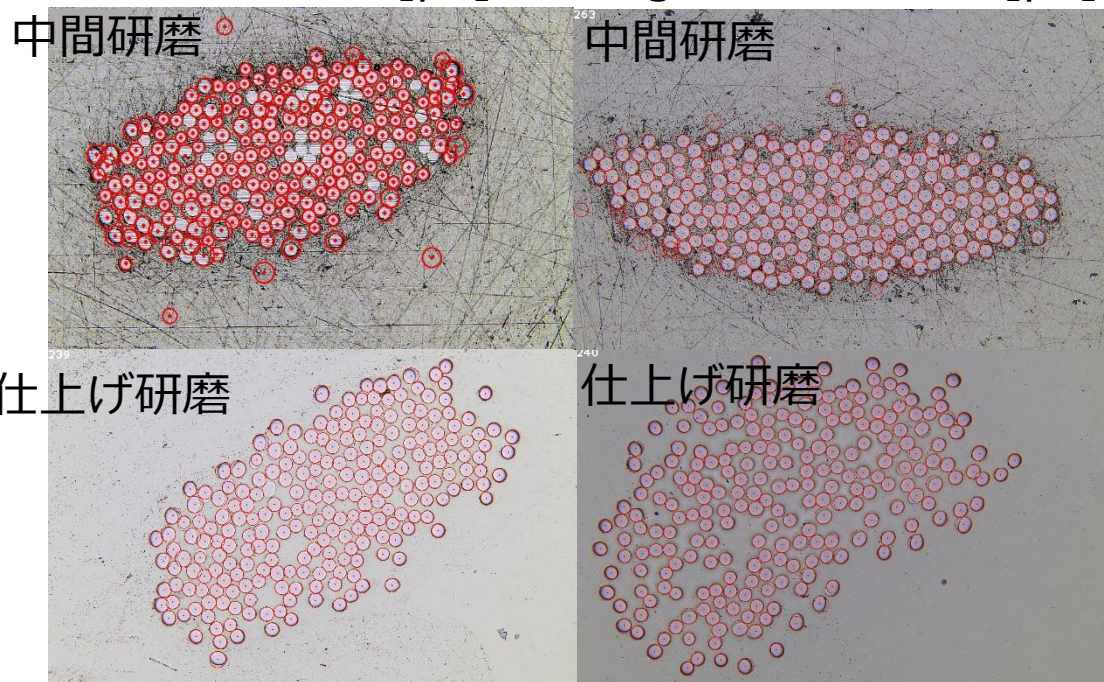
print(circles)

if (circles != None).any():
    circles = np.uint16(np.around(circles))
```

作成したプログラムの流れ

- (1)カーネル処理
- (2)ガウス処理
- (3)グレースケール
- (4)エッジ処理
- (5)円検出

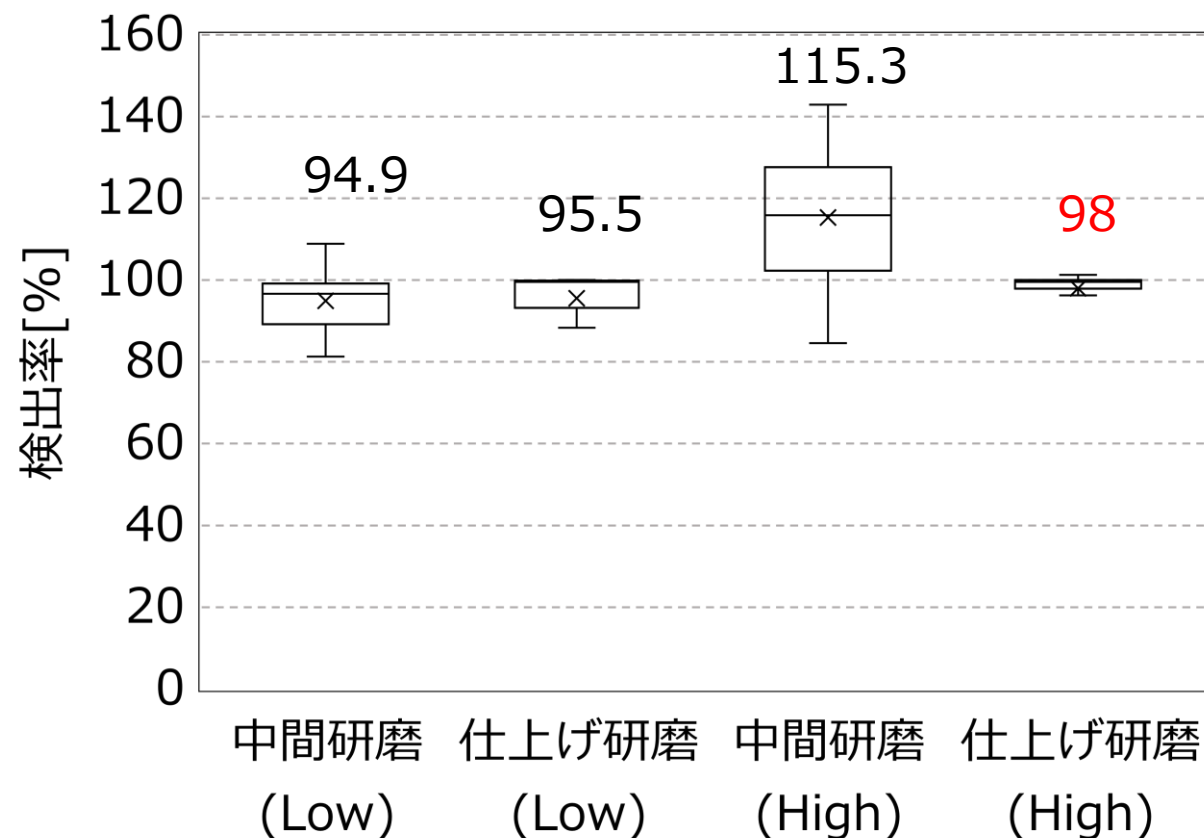
Low:960×600[px] High:2880×1800[px]



検出結果例

画像処理によるフィラメント検出率

○中間研磨及び仕上げ研磨後の画像(10枚以上)からフィラメントを検出、カウントした。



検出率は100に近いほど良い
±4%以内カウント容易

画素数を変えることで、検出率が向上する場合(仕上げ研磨)と、100%を大きく超える場合(中間研磨)があった。ノイズ(傷)の有無による影響と考えられる。

機会学習によるフィラメント数のカウント

○物体検知アルゴリズムの一つであるYOLOを用いてフィラメントを検出カウントするために学習データを作成した。

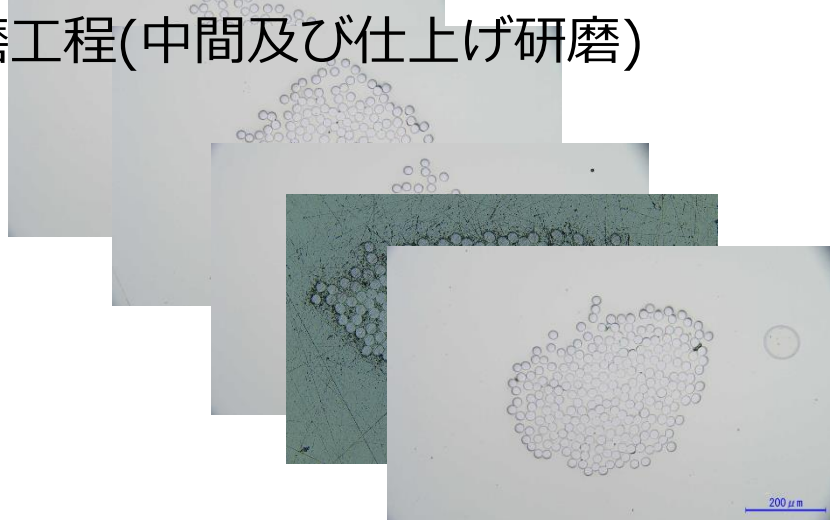
○結果は、検出率で評価した。

$$\text{検出率(\%)} = \frac{\text{(画像内のフィラメント数)}}{\text{(検出したフィラメント数)}} \times 100$$

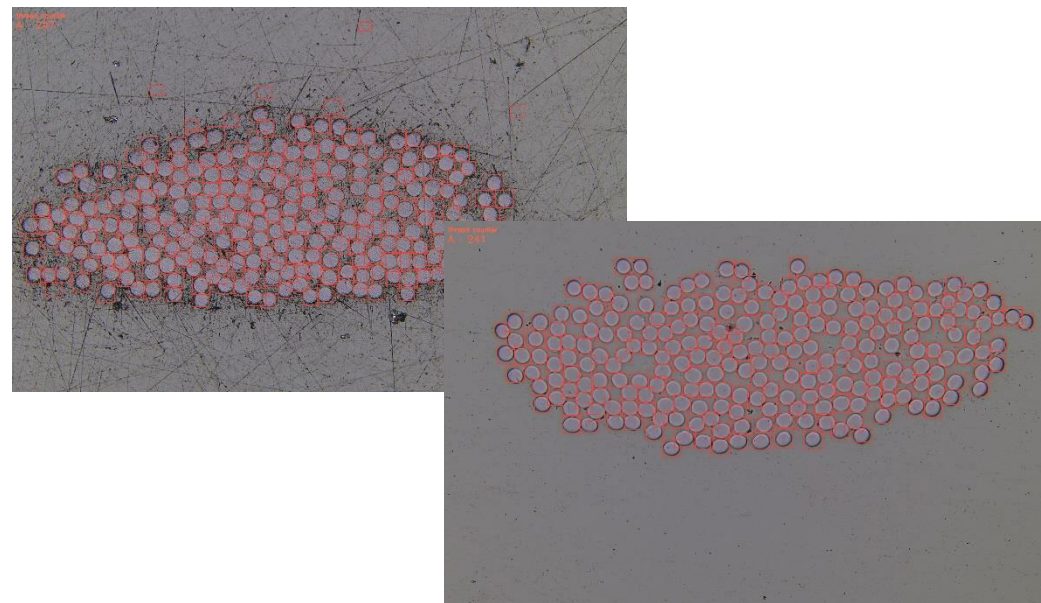
YOLOv5

学習データ数 : 10または25枚

画像 : 濃淡、フィラメント断面形状、研磨工程(中間及び仕上げ研磨)



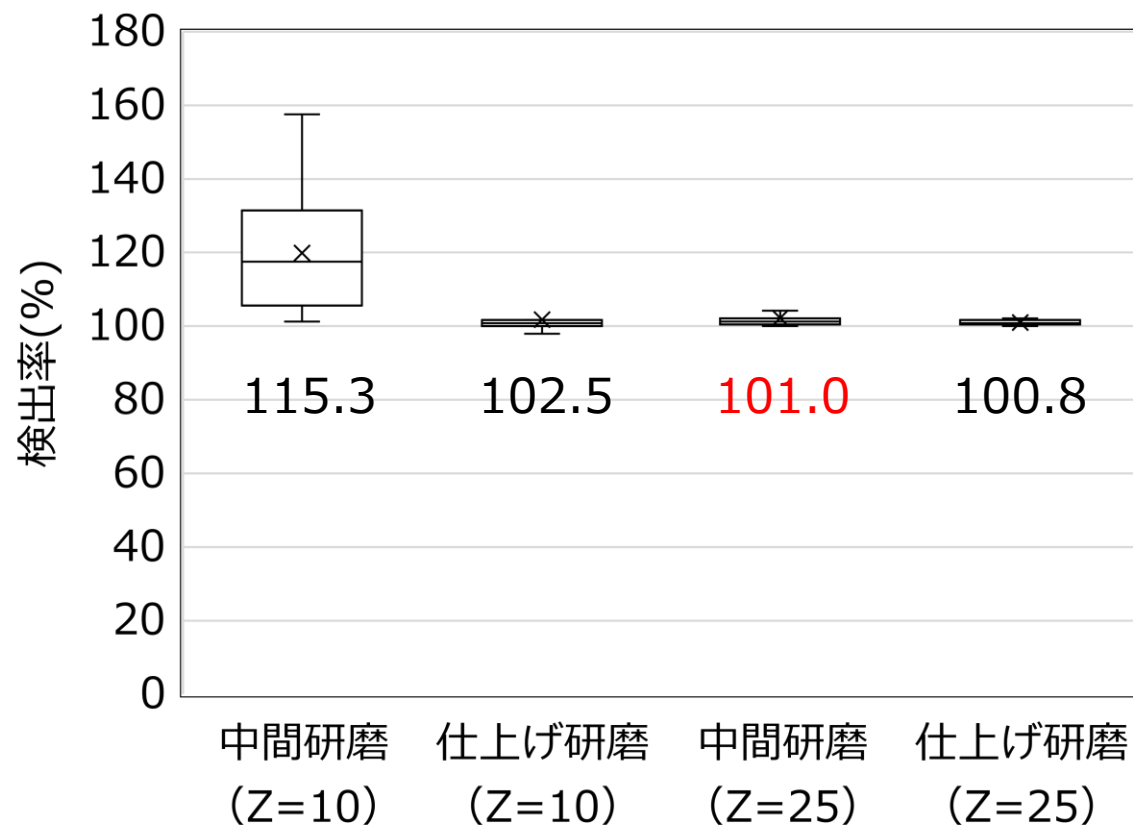
学習、推論



推論結果例

機会学習によるフィラメント検出率

○中間研磨及び仕上げ研磨後の断面画像(10枚以上)から、フィラメント数をカウントした。



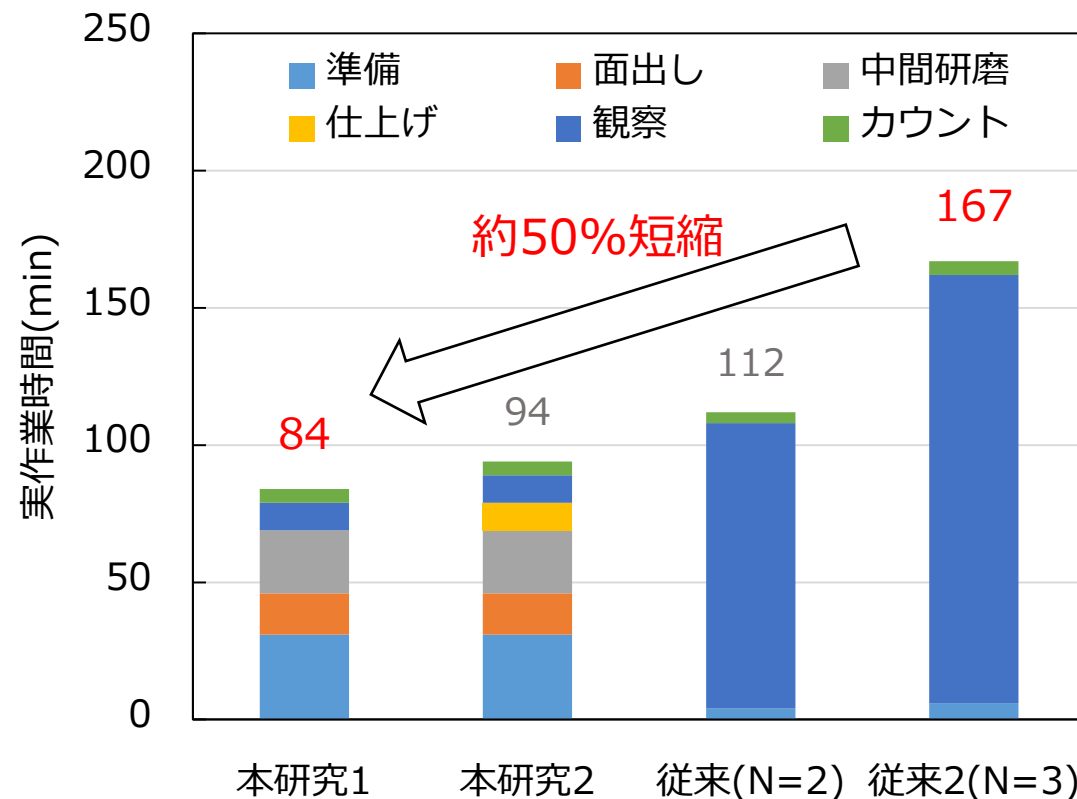
検出率は100に近いほど良い
±4%以内カウント容易

学習データ数を増やすことによって、誤検出を低減しフィラメントをカウントできた。

Z=25では、**中間研磨でも検出率が101.0%**となった。

作業時間の比較

○生地のタテ糸及びヨコ糸のフィラメント数を測定することを想定して2本を3回ずつ測定した場合(N=3)の作業時間と、樹脂包埋した糸を観察する本研究手法(中間研磨または仕上げ研磨まで実施)で作業時間を比較した。



本手法によって、従来の方法よりも作業時間を中間研磨までの場合約50%短縮することができることがわかった。

まとめ・展望

○画像処理の場合、仕上げ研磨まで、機械学習の場合中間研磨まで実施することで高い検出率でフィラメントを検出できた。

○糸を樹脂包埋し、断面画像を観察することで画像処理又は機械学習によってフィラメント数をカウントすることで、従来の方法に比べて作業時間を短縮することができた。

○今後は、フィラメント数をカウントするだけでなく、異形を含む複数の種類が引きそろえられたフィラメントの解析や、延伸加工前後の直径の変化といった物性情報の取得などに活用していきたいと考える。