

Создание карты поверхности цилиндрических труб, отражающей количество пыли в токамаке

Команда “AI Innovators”

Бехтин Артём Владимирович
Суббот Диана Сергеевна
Логвинова Элина Сергеевна
Васильев Дмитрий Владимирович

Москва – 2023

ПРЕДПОСЫЛКИ ПРОЕКТА



Ключевые предпосылки, проблема:

Большое время обработки видеозаписей из токамака человеком.

Решение:

Обработка видеозаписей нейронными сетями позволит сократить время анализ.

Возможность, которую адресует проект:

Наш проект поможет команде инженеров, работающим с ядерным реактором, получать результаты исследований более эффективно.

ЦЕЛИ ПРОЕКТА:

Ожидаемый бизнес-эффект

Снижение времязатрат при анализе количества пыли.

Как следствие:

1. уменьшение показателя чел/ч;
2. увеличение работоспособности сотрудников.

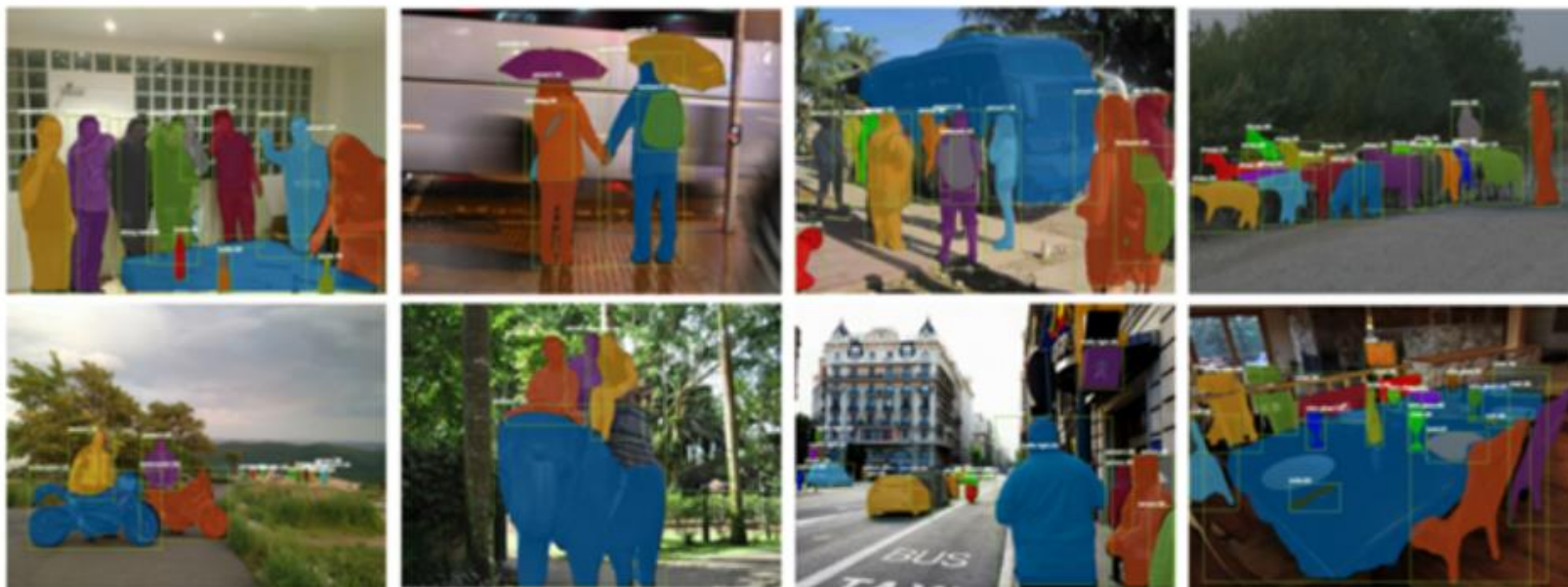
Критерии успеха проекта

1. Автоматизированный процесс поиска пыли, заменяющий работу человека, будет использоваться очень часто.
2. Продукт выгоден команде инженеров МИФИ.

АНАЛИЗ АНАЛОГОВ

Mask Region-based Convolutional Network (Mask R-CNN)

Еще одно расширение модели Faster R-CNN добавленной параллельной ветви к обнаружению ограничивающей рамки, чтобы предсказать маску объекта. Маска объекта — это его сегментация по пикселям на изображении. Эта модель превосходит современную в четырех задачах COCO: сегментация экземпляра, обнаружение ограничивающей рамки, обнаружение объекта и обнаружение ключевой точки.



Примеры применения Mask R-CNN в тестовом наборе данных COCO. Модель определяет каждый объект изображения, его локализацию и точную сегментацию по пикселям. Источник: K. He and al. (2017)


ЭТАПЫ ПРОЕКТА И ТЕХНОЛОГИИ, КОТОРЫЕ МЫ ИСПОЛЬЗОВАЛИ

1. Определение плана работ, распределение ролей. (отв. Бехтин Артем)
2. Подготовка DataSet с помощью OpenCV, matplotlib. (отв. Суббот Диана, Васильев Дмитрий)
3. Обучение модели Faster-RCNN на подготовленных данных. (отв. Бехтин Артем)
4. Написание функции, отвечающей за создание карты, с помощью OpenCV. (отв. Логвинова Элина, Васильев Дмитрий)
5. Тестирование альфа-версии продукта. (отв. Бехтин Артем, Суббот Диана)
6. Исправление недочетов. (отв. Бехтин Артем, Суббот Диана)
7. Тестирование конечного продукта. (отв. Бехтин Артем, Суббот Диана)

ПРОЦЕСС. Подготовка данных

```
main.py
1 import cv2
2 import matplotlib.pyplot as plt
3 import matplotlib
4 matplotlib.use('Qt5Agg')
5
6 video=cv2.VideoCapture("Endoscope video (2).mp4")
7 i=0
8 while video.isOpened():
9     i+=1
10    cap, frame = video.read()
11    if(i%25==0):
12        plt.imshow(frame)
13        plt.show()
14
```

Figure 1



0 100 200 300 400 500 600

docs.google.com/spreadsheets/d/16MaSaDgMqjcdX4QPlesZUiNVUz7EYp4TQO...-cqpVTg/edit#gid=0

data

Файл Правка Вид Вставка Формат Данные Инструменты Расширения ...

100% 123 По ум... 10 B I A

	A	B	C	D	E	F	G	H
	filename	class	x_min	y_min	x_max	y_max	class	x_min
1	Artem_1.png	1	385	383	416	415		
2	Artem_2.png	1	325	273	372	384		
	Artem_3.png	2	294	262	351	475		
	Artem_4.png	2	260	287	333	468		
	Artem_5.png	2	267	267	376	477	1	3
	Artem_6.png	1	403	350	470	398	1	3
	Artem_7.png	1	353	373	453	411		
	Artem_8.png	2	229	254	413	476		
	Artem_9.png	2	327	344	467	476		
	Artem_10.png	2	159	356	561	476		
	Artem_11.png	2	301	297	398	468		
	Artem_12.png	2	195	357	540	468		
	Artem_13.png	2	322	331	368	479		
	Artem_14.png	2	280	358	352	478		
	Artem_15.png	1	344	327	423	398	1	3
	Artem_16.png	1	328	355	407	477		
	Artem_17.png	1	330	325	465	478		
	Artem_18.png	2	349	380	400	452	1	3
	Artem_19.png	2	323	291	450	476		
	Artem_20.png	2	319	385	383	478		
	Artem_21.png	2	353	386	384	444		
	Artem_22.png	3	302	175	438	333		
	Artem_23.png	1	341	376	378	350		
	Artem_24.png	3	305	308	481	449	3	3
	Artem_25.png	1	350	378	385	451		
	Artem_26.png	3	334	249	425	478		
	Artem_27.png	4	278	261	518	349		
29	Diana_26.png	4	260	309	638	475		
30	Dima14.png	3	315	236	501	476		
31	Diana_28.png	4	269	245	605	476		
32	Dima16.png	3	300	280	542	477		
33	Dima17.png	3	321	288	549	472		
34	Dima18.png	3	334	294	486	477		
35	Diana_33.png	3	300	233	460	347		

ПРОЦЕСС. Обучение модели

```
Файл  Изменить  Вид  Вставка  Среда выполнения  Инструменты  Справка  Не удается сохранить изменения с 09:37
+ Код  + Текст  T4  ОЗУ  Диск

[ ] def training(model, train_loader, val_loader, epochs=10):
    # construct an optimizer
    params = [p for p in model.parameters() if p.requires_grad]
    optimizer = torch.optim.SGD(params, lr=0.001, momentum=0.9, weight_decay=0.0005)
    # and a learning rate scheduler
    lr_scheduler = torch.optim.lr_scheduler.StepLR(optimizer,
                                                    step_size=3,
                                                    gamma=0.1)

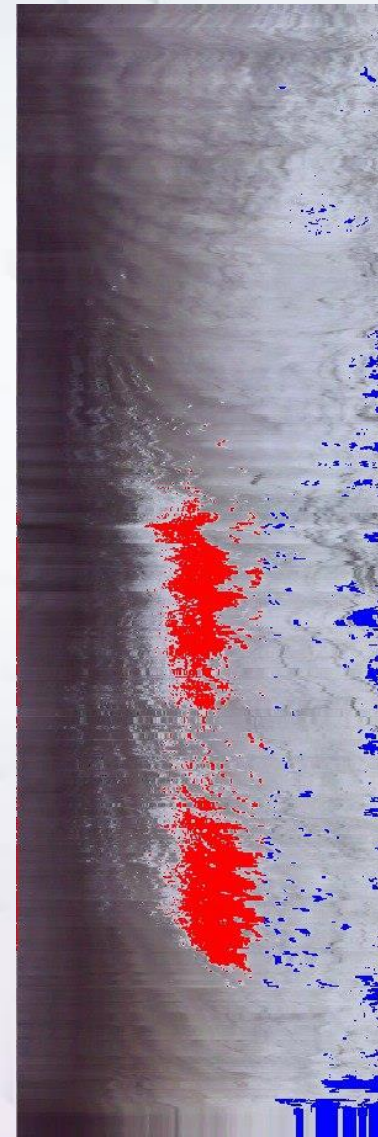
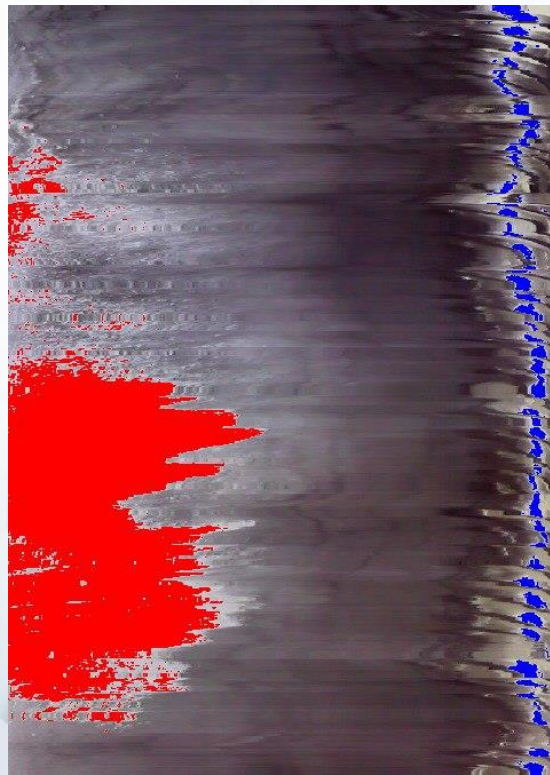
    for epoch in range(epochs):
        # train for one epoch, printing every 10 iterations
        train_one_epoch(model, optimizer, train_loader, device, epoch, print_freq=10)
        # update the learning rate
        lr_scheduler.step()
        # evaluate on the test dataset

▶ training(detection_model, data_loader_train, val_loader=data_loader_train, epochs=10)

[ ] torch.save(detection_model.state_dict(), id+'hack/densenet121_norm.pth')
```


РЕШЕНИЕ. УДОБСТВО ЕГО ИСПОЛЬЗОВАНИЯ

Красным цветом обозначена пыль, синим -
отверстия в трубе.



ИНТЕРФЕЙС ДЛЯ ЗАГРУЗКИ ВИДЕО, КОТОРОЕ НУЖНО ОБРАБОТАТЬ



Dust Map - [Preview] - Qt Designer

Enter the path to the video file of the arrival through the pipe:

Apply

БУДУЩЕЕ ПРОЕКТА. КАК МОЖНО РАЗВИТЬ НАШЕ РЕШЕНИЕ?

1. Обучение модели на большем количестве данных.
2. Использование различных архитектур моделей (использование других методов распознавания объектов).
3. Изучение и использование методов стабилизации изображения с минимизацией потерь.
4. Интеграция проекта в работу инженеров.



**Остались вопросы?
Есть предложения по сотрудничеству?**

GitHub:

email: diasubbot@gmail.com

tg: @AVBekhtin

