

ВСЕРОССИЙСКИЙ КВАНТОВЫЙ ХАКАТОН

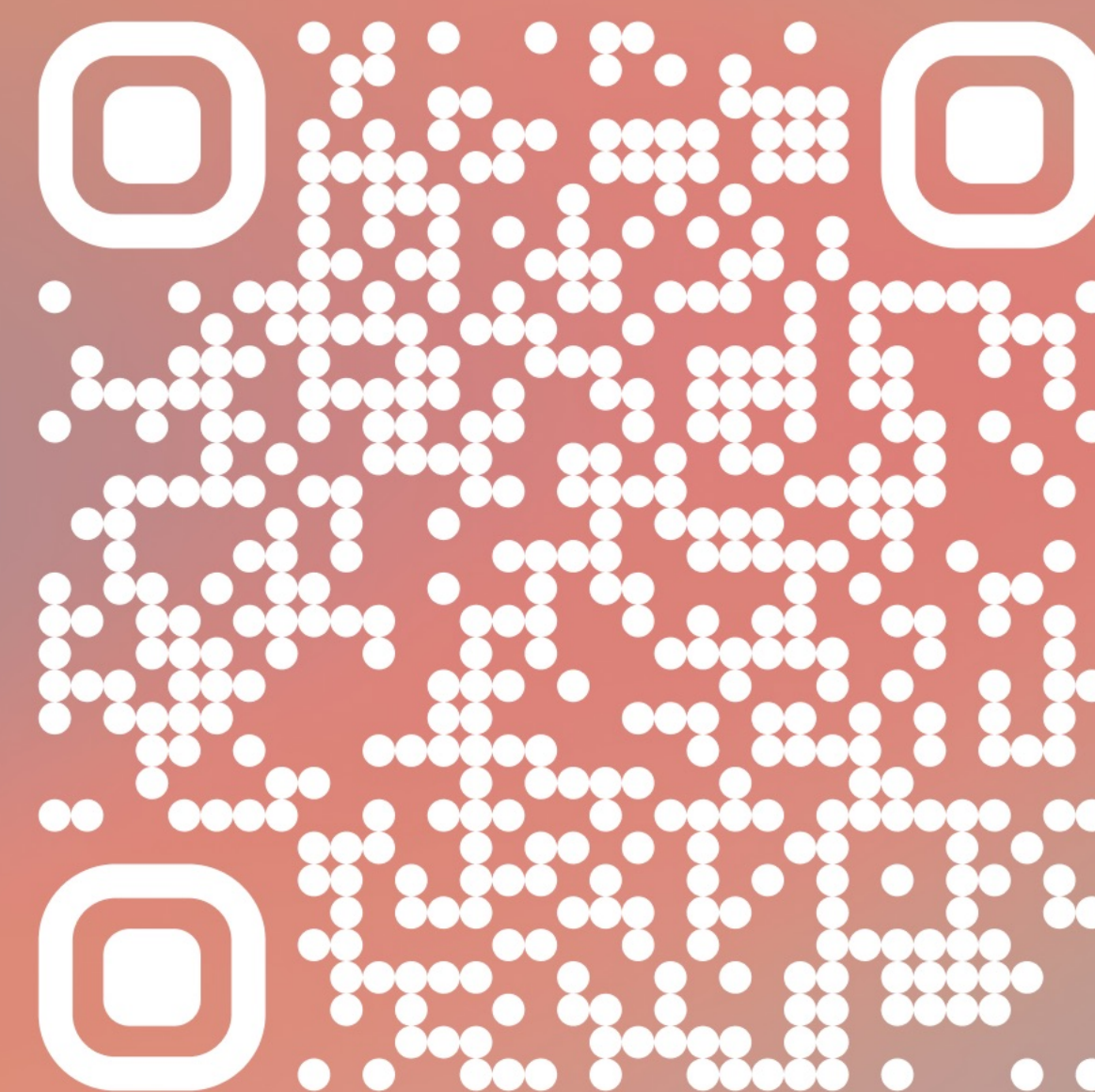
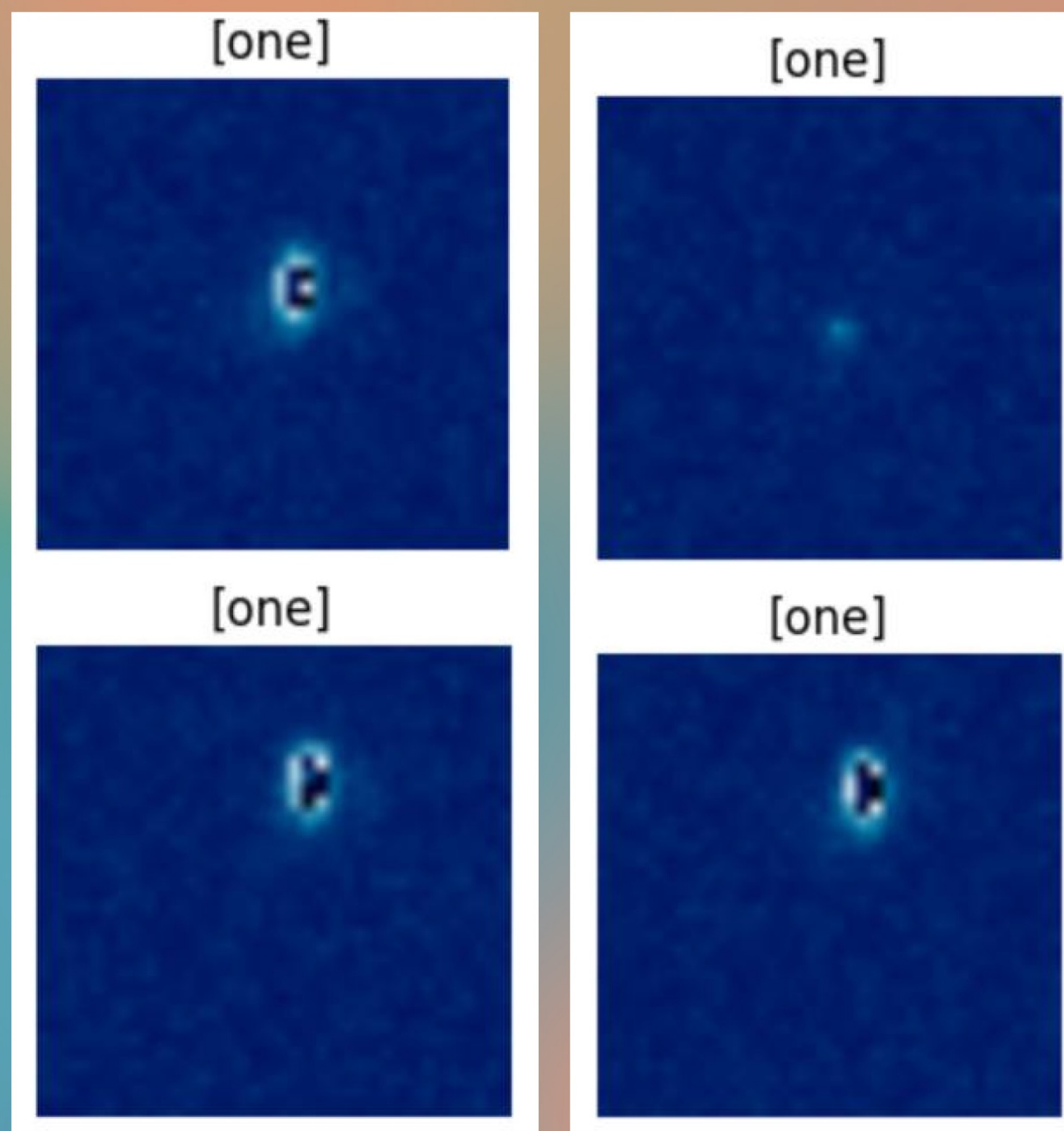
Команда №1 УрФУ “Шапочники”



2 ЗАДАЧА

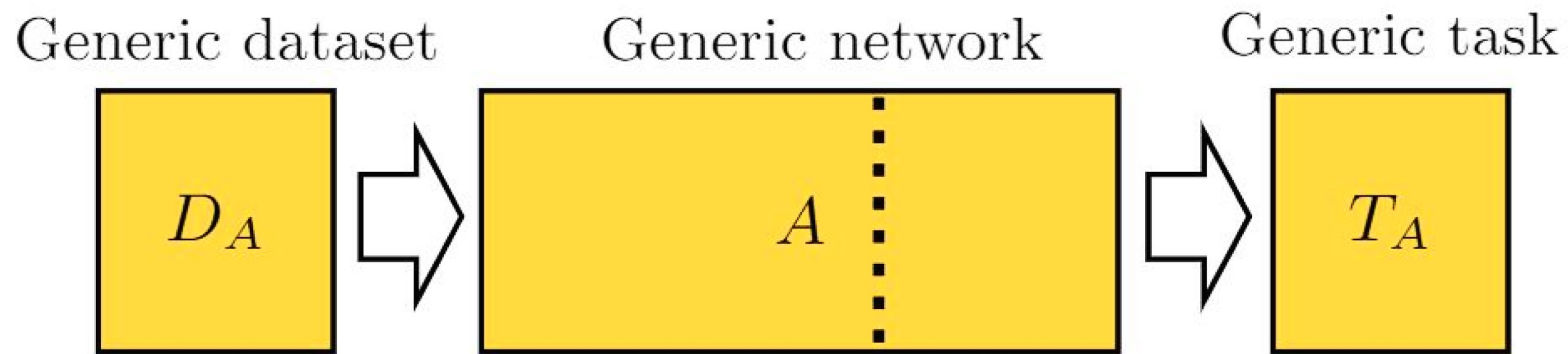
На входе даны 1000 снимков ионов в ловушке Пауля. Изображения даны вертикально. Считать ионы сверху вниз. Входные данные можно скачать по ссылке.

РЕЗУЛЬТАТ

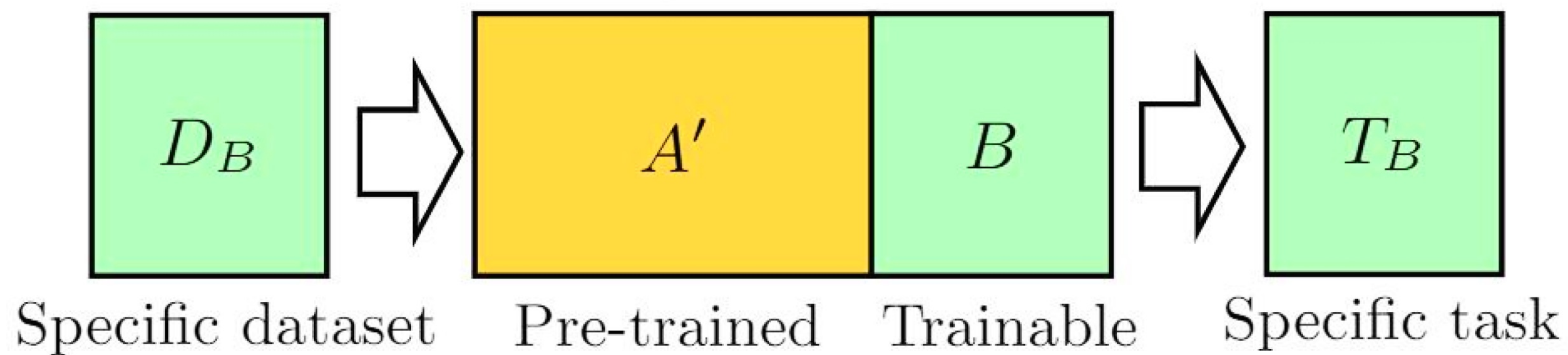


АЛГОРИТМ

- 1** На вход подается исходное изображение (с 4 ионами)
- 2** Изображение копируется, и переводится в «оттенки серого»
- 3** Вычисляется «пороговое значение», пиксели окрашиваются в черный или белый
- 4** Изображение разбивается на 4 секции, в каждом из которых может либо быть, либо не быть ион
- 5** Производится «сверток» изображения, для того, чтобы уменьшить его размер до размерности 1 на 1 пиксель
- 6** Получаем цвет «изображение-пикселя» и отправляем в соответствующую категорию
- 7** «Вырезаем» из оригинального изображения необходимые нам фрагмент с ионом



A'





ResNet18

L
 $512 \rightarrow 4$

QPU

L
 $4 \rightarrow 2$



“bee”

АНАЛИЗ АЛГОРИТМА

```
[39] def score_model(model):  
    model.eval()  
    total_preds = torch.tensor([], dtype=torch.long, device=device)  
    total_labels = torch.tensor([], dtype=torch.long, device=device)  
    with torch.no_grad():  
        for _i, (inputs, labels) in enumerate(dataloaders["validation"]):  
            inputs = inputs.to(device)  
            labels = labels.to(device)  
            outputs = model(inputs)  
            _, preds = torch.max(outputs, 1)  
            total_preds = torch.cat((total_preds, preds), dim=0)  
            total_labels = torch.cat((total_labels, labels), dim=0)  
  
    precision, recall, f1 = calculate_metrics(total_preds, total_labels)  
    print("Precision:", precision)  
    print("Recall:", recall)  
    print("F1:", f1)
```

▶ score_model(model_hybrid)

➤ Precision: 0.9407216494845361
Recall: 1.0
F1: 0.9694555112881806