



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования

«Дальневосточный федеральный университет»  
(ДВФУ)

**ИНСТИТУТ ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ  
ТЕХНОЛОГИЙ**

**ОТЧЁТ ПО ПРАКТИКЕ**

**Учебная ознакомительная практика**

в период с \_\_\_\_\_ по \_\_\_\_\_

В \_\_\_\_\_  
(наименование базы практики)

Выполнил студент М9123-09.04.01 ииБД  
(группа)

\_\_\_\_\_ Сизоненко Г. Г.  
(подпись) (Ф.И.О.)

« \_\_\_\_\_ » \_\_\_\_\_ 2024 г.

Руководитель практики  
от университета

\_\_\_\_\_ Костенков В. А.  
(подпись) (Ф.И.О.)

« \_\_\_\_\_ » \_\_\_\_\_ 2024 г.

Оценка \_\_\_\_\_

Владивосток  
2024

## Оглавление

<b>Введение</b>	<b>3</b>
<b>1 Основная часть</b>	<b>5</b>
1.1 Индивидуальное задание практики	5
1.2 Выбор моделей	5
1.3 YOLO	6
1.4 RT-DETR	7
1.5 Faster R-CNN	9
1.6 Исследование моделей	10
<b>Заключение</b>	<b>18</b>

## **Введение**

Учебная ознакомительная практика в ДВФУ направлена на формирование и развитие первичных профессиональных умений и навыков в сфере избранной специальности, в том числе в области разработки программных продуктов с применением современных информационных технологий с учётом тенденции развития программирования и математического обеспечения.

Учебная ознакомительная практика позволяет систематизировать знания, умения и навыки студента, что обеспечивает становление профессиональных компетенций будущего магистра.

Задачами учебной практики являются:

- приобретение первых практических навыков по выбранному направлению подготовки;
- формирование единства теоретической и практической подготовки, закрепление и углубление полученных теоретических знаний и практических навыков в области разработки программных продуктов с применением современных информационных технологий;
- участие в сборе внутренней и внешней информации и приобретение навыков самостоятельной ее обработки и анализа;
- проверка достоверности собранных данных;
- приобретение и развитие навыков, способствующих формированию творческого подхода в решении проблем проектной и производственно-технологической деятельности;
- сбор фактического материала для подготовки магистерской диссертации: конкретизация направлений магистерского исследования, необходимого объёма информации для обобщения своих знаний по выбранной теме магистерской диссертации.

Содержание и программа практики:

- ознакомление с программой, местом и временем проведением практики;
- ознакомление с литературными источниками по теме практики;

- проведение исследования информационной базы индивидуального проекта;
- выполнение индивидуального проекта практики;
- подготовка отчёта по практике и его защита.

## **1 Основная часть**

### **1.1 Индивидуальное задание практики**

В качестве индивидуального задания практики было поручено исследовать возможность распознавания медведей при помощи методов машинного обучения.

Само по себе распознавание медведей является актуальной и интересной задачей, так как разработанные решения можно использовать для:

- Исследование поведения медведей
- Обеспечение безопасного проведения работ в областях, которые расположены в местах обитания медведей
- Исследование миграции медведей
- Наблюдение за медведями в местах их содержания(вольеры, заповедники и т.д.)

### **1.2 Выбор моделей**

Для выполнения задания в первую очередь требовалось выбрать рассматриваемые модели. К моделям были применены следующие требования, а именно:

- Быстродействие
- Точность
- Возможность решать задачи классификации изображений или детекции

В частном случае задача детекции была превращена в задачу классификации, так как если модель смогла на изображении найти объект с нужным классом, то в качестве ответа получали только метку класса, а именно есть медведь на изображении или он отсутствует.

Были выбраны следующие модели:

- YOLO
- RT-DETR
- Faster R-CNN

Для всех моделей были использованы веса, которые обучили на датасете Common Objects in Context (COCO).

### 1.3 YOLO

Когда в задаче говорится о детекции объектов, то в первую очередь для решения этой задачи в голову приходит архитектура YOLO.

YOLO (You Only Look Once), популярная модель обнаружения объектов и сегментации изображений, была разработана Джозефом Редмоном и Али Фархади в Университете Вашингтона. Появившись в 2015 году, YOLO быстро завоевала популярность благодаря своей высокой скорости и точности.

На сегодняшний день существует не менее 10 версий описываемой архитектуры.

Алгоритм YOLO, был первой попыткой сделать возможной детекцию объектов в реальном времени. В рамках алгоритма YOLO исходное изображение сначала разбивается на сетку из  $N \times N$  ячеек. Если центр объекта попадает внутрь координат ячейки, то эта ячейка считается ответственной за определение параметров местонахождения объекта. Каждая ячейка описывает несколько вариантов местоположения ограничивающих рамок для одного и того же объекта. Каждый из этих вариантов характеризуется пятью значениями — координатами центра ограничивающей рамки, его шириной и высотой, а также степени уверенности в том, что ограничивающая рамка содержит в себе объект. Также необходимо для каждой пары класса объектов и ячейки определить вероятность того, что ячейка содержит в себе объект этого класса. Таким образом, последний слой сети, принимающий конечное решение об ограничивающих рамках и классификации объектов работает с тензором размерности  $N \times N \times (5B + C)$ , где  $B$  — количество предсказываемых ограничивающих рамок для ячейки,  $C$  — количество классов объектов, определённых изначально.

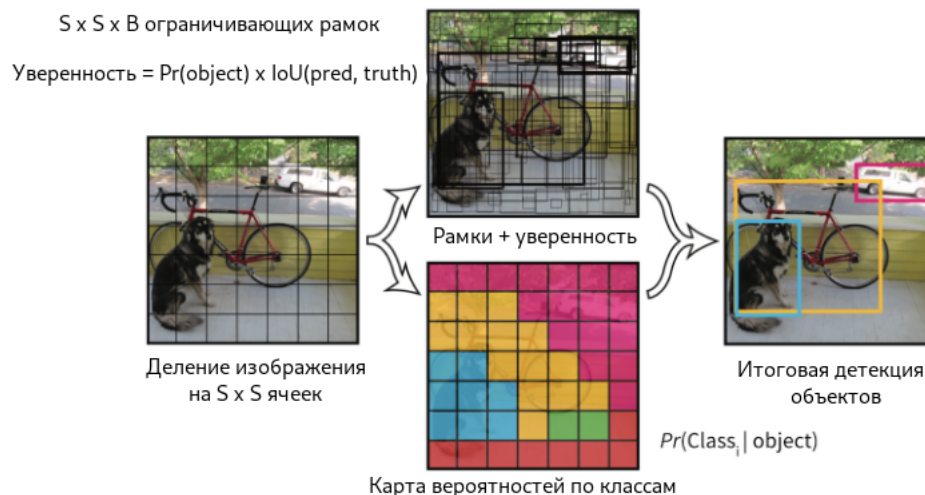


Рисунок 1 – Алгоритм YOLO

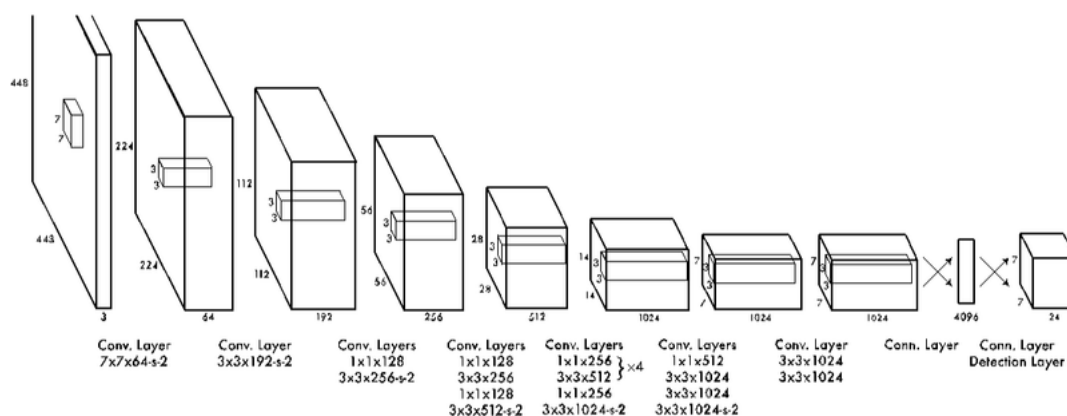


Рисунок 2 – Архитектура YOLO

## 1.4 RT-DETR

Следующей исследуемой архитектурой стали визуальные трансформеры. К использованию Real-Time Detection Transformer(RT-DETR) подтолкнула статья на arxiv "DETRs Beat YOLOs on Real-time Object Detection" и готовая реализация в библиотеке ultralytics.

По мнению авторов статьи, использование подавления немаксимумов (Non-Maximum Suppression, NMS), который является важным шагом в детекции YOLO, негативно сказывается на производительности и точности модели. В свою очередь авторы отказываются от NMS и описывают свою архитектуру следующим образом.

Мы передаем признаки из последних трех слоев бэкбона в кодер.

Эффективный гибридный кодер преобразует многомасштабные признаки в последовательность признаков изображения посредством внутримасштабного взаимодействия признаков основанного на внимании (Attention based Intra-scale Feature Interaction, AIFI) и слияния межмасштабных признаков на основе CNN (CNN-based Cross-scale Feature Fusion, CCFF). Затем выбор запроса с минимальной неопределенностью выбирает фиксированное количество признаков кодера, которые будут служить начальными объектными запросами для декодера. Наконец, декодер со вспомогательными головами прогнозирования итеративно оптимизирует объектные запросы для генерации категорий и блоков.

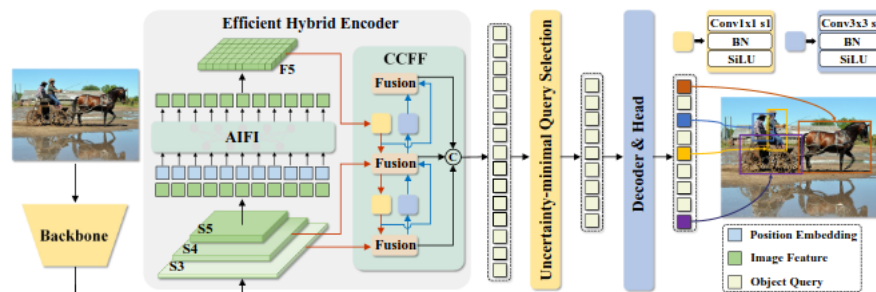


Рисунок 3 – Представление RT-DETR

Также авторы сделали бенчмарк с датасетом MS COCO и сравнили свою архитектуру с другими моделями детекции в реальном времени и показали, что их архитектура достигает SOTA

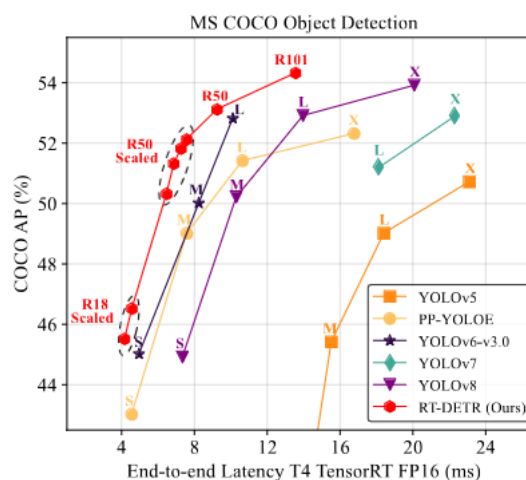


Рисунок 4 – Сравнение RT-DETR с другими моделями детекции



## 1.5 Faster R-CNN

Заключительной рассматриваемой архитектурой стали Сверточные сети, основанные на регионах или же Region-based CNN (R-CNN), а в частности их модификация Faster R-CNN.

Эта архитектура была выбрана для рассмотрения как одна из классических архитектур, которая решает задачу детекции. В её основе лежит алгоритм выборочного поиска (Selective Search). Данный алгоритм получает на вход изображение, а на выходе выдает массив прямоугольников, в которых возможно находится объект. И идея такова, что в первую очередь будет проведен выборочный поиск, при помощи которого мы получим кропы изображений, а уже эти кропы мы помещаем в классификатор.

В дальнейшем эту идею развили, потому что нам не нравилось, что для каждой гипотезы из выборочного поиска мы выполняли классификацию, поэтому решили сперва выполнить свертку и получить карту признаков (Feature Map), затем на исходном изображении получить гипотезы и их спроецировать на карту признаков. Этот алгоритм называли Fast R-CNN.

Заключительной идеей в R-CNN стало следующее: выборочный поиск решили заменить на нейронную сеть. И теперь поиск границ объектов это тоже обучаемый алгоритм.

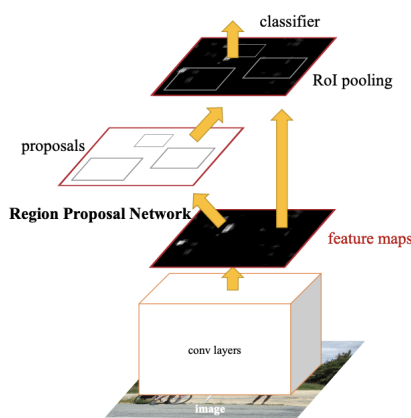


Рисунок 5 – Принцип работы Faster R-CNN

## 1.6 Исследование моделей

Для исследования моделей был собран небольшой тестовый датасет из 71 изображения. 51 изображение содержало медведей, 20 изображений медведей не содержали. Изображения подбирались с учетом специфики задания, а именно подбирались изображения из дикой природы, полученные при помощи фотоловушек или съемки издалека, то есть художественные снимки заведомо не подходили, так как имели высокое качество, что в свою очередь не соответствовало специфике задания.

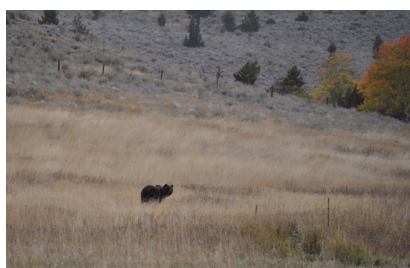


Рисунок 6 – Пример изображения, которое содержит медведя



Рисунок 7 – Пример изображения, которое медведя не содержит

Для выполнения работы были использованы следующие библиотеки:

- ultralytics
- torchvision
- matplotlib
- sklearn

Метрикой качества модели была выбрана ассурасу, также были построены матрицы ошибок и ROC кривая.

Вычисление метрики, просмотрение матриц ошибок и ROC кривой было выполнено в виде функций, в которые передавались истинные метки и метки предказанные моделями. В функцию для построения ROC кривой вместо предсказанных меток передавалась уверенность модели в ответе.

Вычисление точности:

```
def print_accuracy(y_pred_dict, y_true, save = False, file_name
↪ = 'accuracy.txt'):
    file = open(file_name, 'w') if save else None

    for model, y_pred in y_pred_dict.items():
        print(f'{model} acc: {(y_pred == y_true).mean()}',
↪ file=file)

    if save:
        print(f'accuracy metrics saved in {file_name}')
        file.close()
        \caption{Вычисление точности}
```

Построение матрицы ошибок:

```
def plot_conf_mat(y_pred_dict, y_true, rows_cols: tuple[int] |
↪ None = None, figsize: tuple[int] | None = None, save =
↪ False, file_name = 'ConfMat_plot.png'):
    if not rows_cols:
        rows, cols = factor(len(y_pred_dict))
        ↪ #определяем кол-во строк и столбцов при
        ↪ помощи разложения на множители
    rows, cols = rows_cols
```

```

fig, axs = plt.subplots(rows, cols, figsize=figsize)
counter = 0

    for i in range(rows):
        for j in range(cols):
            model, y_pred =
                ↪ list(y_pred_dict.items())[counter]
            cm = confusion_matrix(y_true, y_pred)
            axs[i][j].set_title(model)
            ConfusionMatrixDisplay(cm).plot(ax=axs[i][j])
            counter +=1

plt.tight_layout()

if save:
    plt.savefig(file_name)

```

## Построение ROC кривой

```

def plot_roc_curve(y_conf_dict, y_true, save = False, file_name
    ↪ = 'roc_curve.png'):
    fig, ax = plt.subplots(1, 1, figsize=(10, 10))

    for name, y_conf in y_conf_dict.items():
        fpr, tpr, thresholds = metrics.roc_curve(y_true,
            ↪ y_conf)
        roc_auc = metrics.auc(fpr, tpr)
        metrics.RocCurveDisplay(fpr=fpr, tpr=tpr,
            ↪ roc_auc=roc_auc,
            ↪ estimator_name=name).plot(ax=ax)

    ax.plot((0, 1), (0, 1),
        label="Chance level (AUC = 0.5)",

```

```

        color="k",
        linestyle="--",)
ax.legend()
ax.set_title("Receiver Operating Characteristic (ROC)
→ curves")
ax.grid(linestyle="--")
plt.tight_layout()

if save:
    plt.savefig(file_name)

```

Результаты исследования моделей представлены ниже.

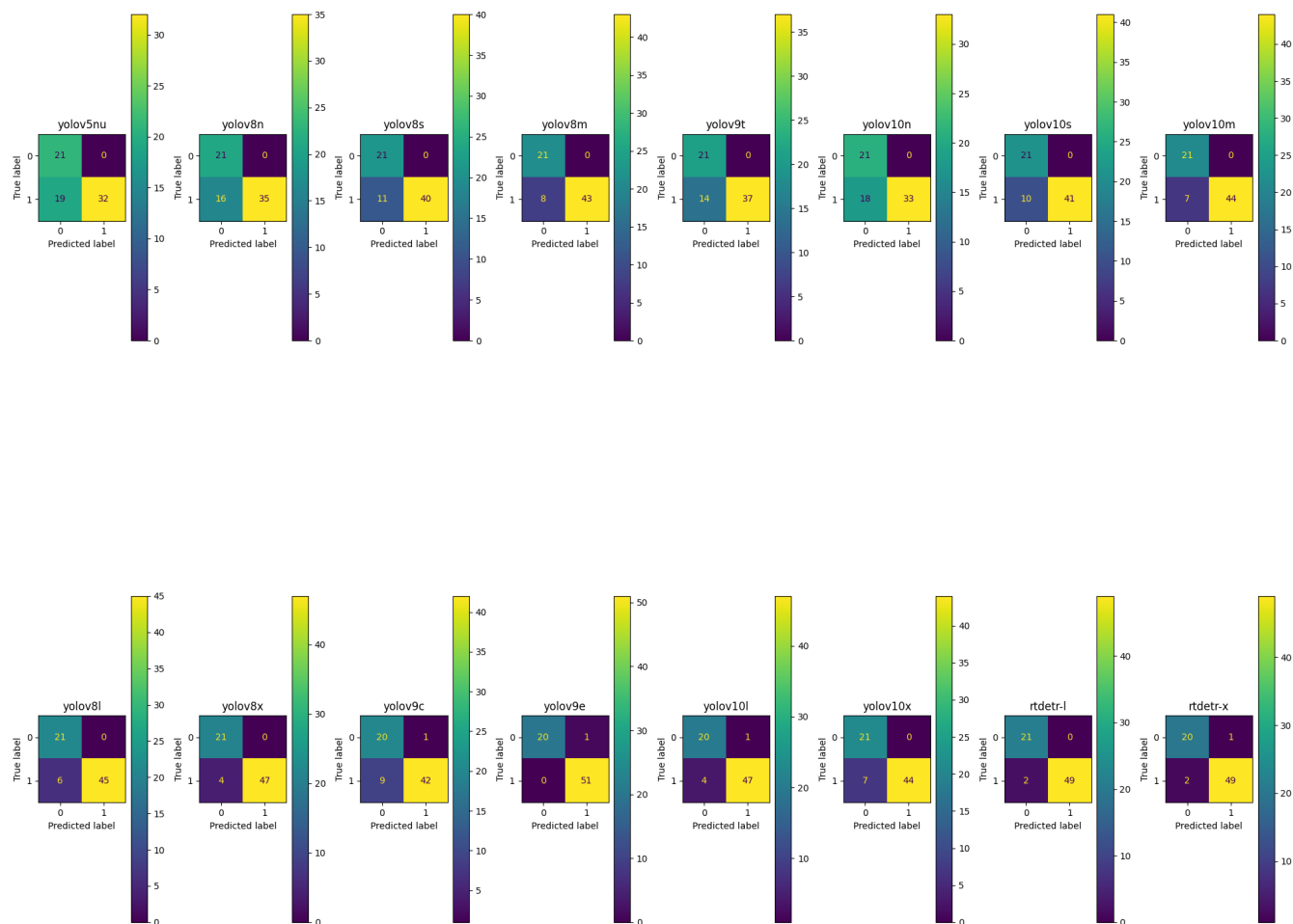


Рисунок 8 – Матрица ошибок для моделей YOLO и RT-DETR

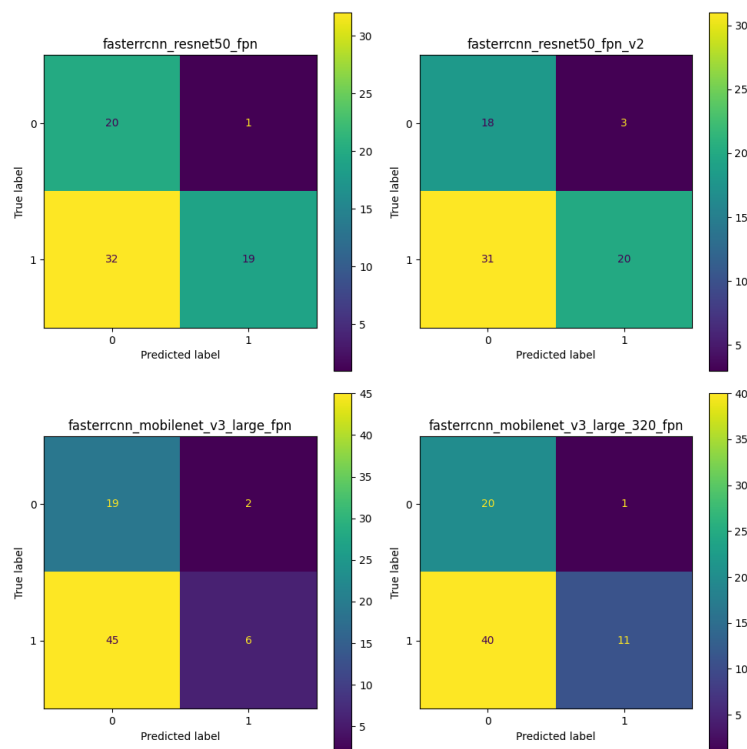


Рисунок 9 – Матрица ошибок для моделей Faster R-CNN

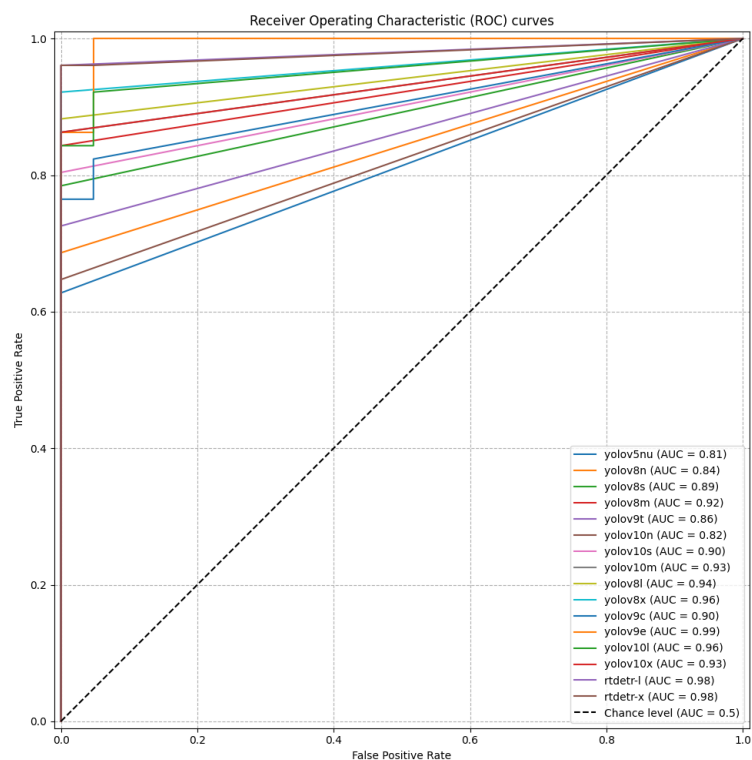


Рисунок 10 – ROC кривая для моделей YOLO и RT-DETR

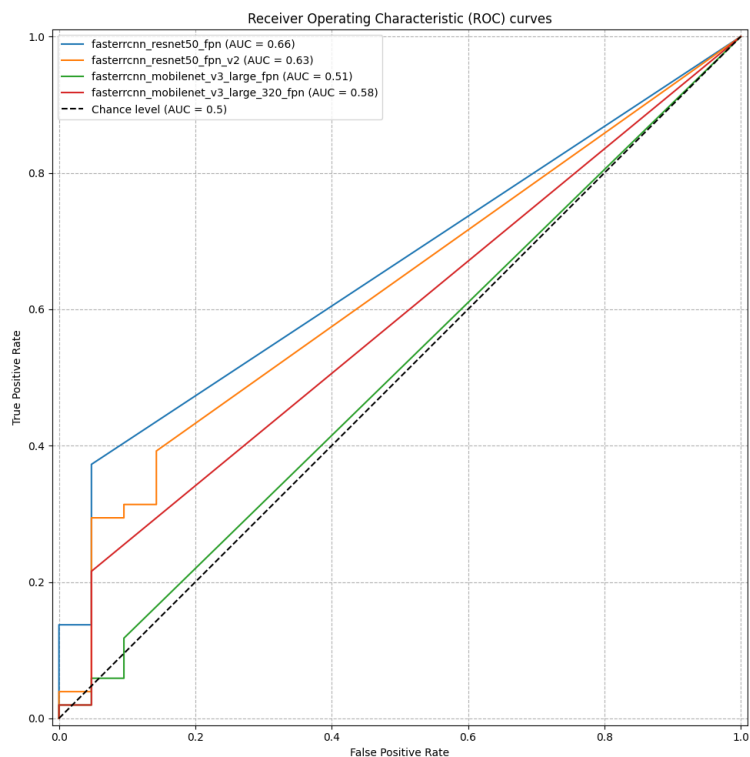


Рисунок 11 – ROC кривая для моделей Faster R-CNN

Полученные значения метрики качества:

```

yolov5nu acc: 0.7361111111111112
yolov8n acc: 0.7777777777777778
yolov8s acc: 0.8472222222222222
yolov8m acc: 0.8888888888888888
yolov9t acc: 0.8055555555555556
yolov10n acc: 0.75
yolov10s acc: 0.8611111111111112
yolov10m acc: 0.9027777777777778
yolov8l acc: 0.9166666666666666
yolov8x acc: 0.9444444444444444
yolov9c acc: 0.8611111111111112
yolov9e acc: 0.9861111111111112
yolov10l acc: 0.9305555555555556
yolov10x acc: 0.9027777777777778

```



rt detr-l acc: 0.9722222222222222  
rt detr-x acc: 0.9583333333333334  
fasterrcnn\_resnet50\_fpn acc: 0.5416666666666666  
fasterrcnn\_resnet50\_fpn\_v2 acc: 0.5277777777777778  
fasterrcnn\_mobilenet\_v3\_large\_fpn acc: 0.3472222222222222  
fasterrcnn\_mobilenet\_v3\_large\_320\_fpn acc: 0.4305555555555556

## Заключение

По полученным результатам исследования можно сделать вывод, что модель yolov9e является наиболее точной в сравнении с другими исследованными моделями. Также эта модель имеет наибольшую площадь под кривой ROC. Изучив матрицу ошибок по этой модели можно обратить внимание, что у модели не было ложно-отрицательных срабатываний. Это означает, что модель не пропустила ни одного изображения содержащего медведя, и в случае когда речь идет о жизнях людей это имеет решающий фактор. Неплохой результат показали RT-DETR, которые уступили лишь одной модели в точности.

Также модели исследовались на синтетических данных в качестве тестовых. Это была созданная сцена с медведем в лесу. Основная проблема с которой столкнулись исследуемые модели заключалась в том, что не получалось обнаружить медведя на этих изображениях. Эту проблему удавалось нивелировать дообучением исследованных моделей на синтетических данных, что могло бы помочь в задаче детекции медведей в специфической местности.

Во время прохождения практики были успешно выполнены все поставленные задачи. Получены первичные профессиональные умения и навыки в области разработки программных продуктов с применением современных информационных технологий, сформировано единство теоретической и практической подготовки, закреплены теоретические знания и практические умения, был получен опыт самостоятельной сборки, анализа и проверки внешней и внутренней информации, сформированы навыки творческого решения проблем проектной и производственно-технологической деятельности.

При выполнении индивидуального практического задания со стороны руководителя практики и преподавателей была оказана значительная помощь в поиске и изучении нового материала, выполнении заданий, оформлении отчёта.



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего образования

**«Дальневосточный федеральный университет»  
(ДВФУ)**

**ИНСТИТУТ ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ  
ТЕХНОЛОГИЙ**

**ДНЕВНИК  
Прохождения практики  
Учебная ознакомительная практика**

Студент \_\_\_\_\_ Сизоненко Григорий Григорьевич  
Группа \_\_\_\_\_ М9123-09.04.01иибд

Владивосток  
2024

<b>Дата выполнения работ</b>	<b>Место</b>	<b>Краткое содержание выполняемых работ</b>	<b>Отметка о выполнении работы</b>
	ЦПИР	Получение индивидуального практического задания	
	ЦПИР	Выполнения индивидуально практического задания	
	ЦПИР	Подготовка отчёта по практике	
	ЦПИР	Защита отчёта	

Руководитель практики от университета

Костенков В. А., инженер ЦПИР

---

(Ф.И.О., должность, подпись)

## Индивидуальное задание по практике

Студенту группы М9123-09.04.01иибд Сизоненко Григорий Григорьевич  
(группа) (Ф.И.О.)

Место прохождения практики: Центр прикладных исследований и разработок

Сроки прохождения практики с \_\_\_\_\_ по \_\_\_\_\_ 2024 года

Виды работ и требования по их выполнению: Ознакомиться с моделями распознавания. Исследовать эти модели в задаче распознавания медведей. Исследуемые модели должны быть протестированы на данных, которые отражают применяемое оборудование для наблюдения за дикими животными.

Руководитель практики от ДВФУ

<u>инженер ЦПИР</u>	<u>Костенков В. А.</u>
(должность)	(подпись) (Ф.И.О.)

«\_\_\_\_» \_\_\_\_\_ 2024 г.