



Министерство науки и высшего образования  
Российской Федерации

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Московский государственный технический университет  
имени Н. Э. Баумана (национальный  
исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

Факультет «Фундаментальные науки»

Кафедра «Математическое моделирование»

## Расчётно-пояснительная записка

к курсовой работе

на тему

## Семантическая сегментация изображений рыб

Студент группы ФН12-71Б

\_\_\_\_\_ А.А. Сысоев  
(подпись, дата)

Руководитель курсовой работы

\_\_\_\_\_ Д.А. Фетисов  
(подпись, дата)

Министерство науки и высшего образования  
Российской Федерации

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Московский государственный технический университет  
имени Н. Э. Баумана (национальный  
исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ

Заведующий кафедрой \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

## ЗАДАНИЕ на выполнение курсовой работы

по дисциплине

Студент группы ФН12-71Б

\_\_\_\_\_  
(Фамилия, Имя, Отчество)

Тема курсовой работы: **Семантическая сегментация изображений рыб**

Направленность КР \_\_\_\_\_  
(учебная, исследовательская, практическая, производственная и др.)

Источник тематики \_\_\_\_\_

График выполнения работы: 25% к \_\_\_\_ нед., 50% к \_\_\_\_ нед., 75% к \_\_\_\_ нед., 100% к \_\_\_\_ нед.

**Задание** \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

### **Оформление курсовой работы:**

Расчётно-пояснительная записка на \_\_\_\_ листах формата А4.

Перечень графического (иллюстративного) материала \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Дата выдачи задания « \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Руководитель курсовой работы \_\_\_\_\_ *Д.А. Фетисов*  
(подпись, дата)

Студент \_\_\_\_\_ *А.А. Сысоев*  
(подпись, дата)

**Примечание.** Задание оформляется в двух экземплярах: один выдаётся студенту, второй хранится на кафедре.

## Содержание

1. Введение . . . . .	4
2. Основная часть . . . . .	5
2.1. Математическая формулировка задачи . . . . .	5
2.2. Архитектура нейронной сети U-Net . . . . .	6
2.3. Подготовка данных . . . . .	9
2.4. Метрики оценки качества модели . . . . .	10
2.5. Функция потерь . . . . .	11
2.6. Результаты эксперимента . . . . .	11
3. Заключение . . . . .	13
4. Список литературы . . . . .	14
5. Приложение . . . . .	15

## 1. Введение

В настоящее время сегментация изображений является одной из основных задач компьютерного зрения. Она показала высокую эффективность благодаря развитию методов глубокого обучения, в частности сверточных нейронных сетей. Сегментация изображений применяется в автономном вождении для распознавания элементов дорожной сцены, в медицинской визуализации для выделения органов и патологий, а также при анализе спутниковых изображений для классификации земной поверхности. Кроме того, семантическая сегментация используется в сельском хозяйстве и промышленности для автоматизации процессов мониторинга и анализа изображений.

В рамках данной работы рассматривается задача бинарной семантической сегментации изображений рыб, где требуется выделить области, соответствующие изображениям рыб.

**Цель работы** — разработка и обучение модели нейронной сети, способной решать задачу семантической сегментации изображений рыб.

Для достижения поставленной цели используются следующие задачи:

- Исследование архитектуры нейронной сети U-net.
- Подготовка и предварительная обработка набора изображений.
- Обучение модели на выбранном датасете.
- Оценка качества работы модели с использованием различных метрик.
- Анализ полученных результатов.

Решение поставленной задачи позволит исследовать эффективность применения архитектуры U-net для сегментации изображений объектов. Полученные результаты могут быть использованы при дальнейшем изучении методов семантической сегментации.

## 2. Основная часть

### 2.1. Математическая формулировка задачи

В общем случае задача сегментации – разделить изображение на фрагменты (группы пикселей) по некоторому общему критерию. В зависимости от критерия получаются разные задачи сегментации изображения.

1. Извлечение объекта – выделение конкретного произвольного объекта, указанного пользователем или по-другому заданного. Например, пользователь может выделить объект ограничивающим прямоугольником или нарисовать контур объекта.
2. Сегментация без учителя – разделение изображения на регионы, однородные по своим визуальным характеристикам и отличающиеся от соседних регионов. Например, нужно выделить области, соответствующие разным фрагментам на одежде человека

Поскольку такие методы разбивают объекты обычно на несколько сегментов, то их еще называют методами пересегментации, потому что мы сегментируем чрезмерно по сравнению с тем, как хотелось бы. Отсюда приходим к понятию семантическая сегментация. **Семантическая сегментация** – попиксельная разметка изображения, где каждая метка соответствует определенному классу объекта [1]. При этом:

1. Разные пиксели одного объекта существенно отличаются друг от друга по признакам (яркости, цвету, текстуре окрестности).
2. Единственное, что у них общее – «семантика»

Для формализации введем следующие обозначения:

- $X = \{x_i\}_{i=1}^N$  – множество пикселей входного изображения;
- $Y = \{y_i\}_{i=1}^N$  – множество истинных меток пикселей;
- $\hat{Y} = \{\hat{y}_i\}_{i=1}^N$  – множество предсказанных меток;
- $y_i, \hat{y}_i \in \{0, 1\}, \quad i = 1, \dots, N.$

Задача семантической сегментации заключается в нахождении такого отображения

$$f_{\theta} : X \rightarrow \hat{Y}, \quad (2.1)$$

где  $f_{\theta}$  – нейронная сеть с параметрами  $\theta$ , которая для каждого пикселя входного изображения предсказывает его принадлежность к соответствующему классу.

Обучение модели сводится к подбору параметров  $\theta$ , минимизирующих функцию потерь между предсказанными метками  $\hat{Y}$  и истинной разметкой  $Y$  на обучающей выборке.

## 2.2. Архитектура нейронной сети U-Net

Для решения задачи семантической сегментации изображений в данной работе используется архитектура **U-Net**, которая доказала свою высокую эффективность в попиксельной разметке объектов [2]. Архитектура сочетает сжимательную и восстанавливающую части, обеспечивая точное выделение объектов даже при ограниченном количестве обучающих данных.

Основным строительным блоком U-Net являются **сверточные слои**. Каждый сверточный слой применяет несколько фильтров размером  $3 \times 3$ , которые скользят по входному изображению и формируют карты признаков. Операция свёртки может быть записана следующим образом:

$$O_{x,y} = \sum_{i,j} W_{i,j} I_{x+i,y+j}, \quad (2.2)$$

где:

- $I$  — входные данные,
- $O$  — выходные данные,
- $W$  — ядро свёртки,
- $(i, j)$  пробегает все элементы ядра свёртки.

Каждый фильтр выделяет определённые локальные признаки, такие как границы, текстуры и углы, позволяя сети постепенно изучать сложные визуальные характеристики.

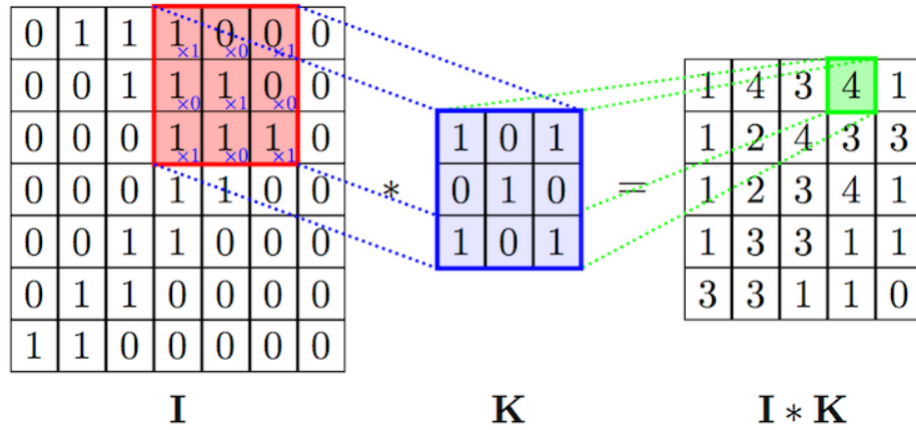


Рис. 2.1. Операция свертки

После свёртки используется **функция активации ReLU**, определяемая как:

$$f(x) = \max(0, x), \quad (2.3)$$

которая добавляет нелинейность в модель, позволяя нейросети аппроксимировать сложные функции. ReLU также предотвращает затухание градиента и ускоряет обучение.

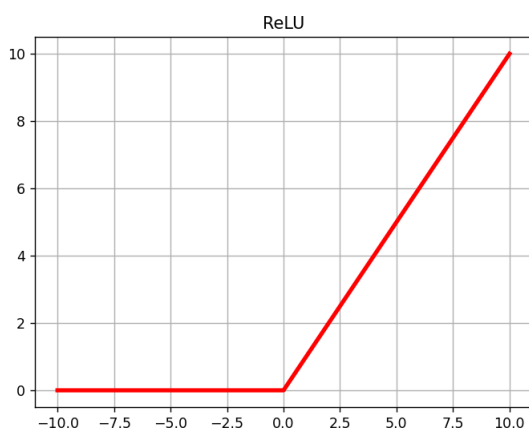


Рис. 2.2. Функция активации

Используется **padding**, чтобы сохранить пространственные размеры карты признаков.

Для уменьшения размерности карты признаков и увеличения области восприятия используется операция **max-pooling**. Обычно применяются окна  $2 \times 2$ , из которых выбирается максимальное значение. Это позволяет сети выделять более глобальные признаки, одновременно снижая вычислительные затраты.

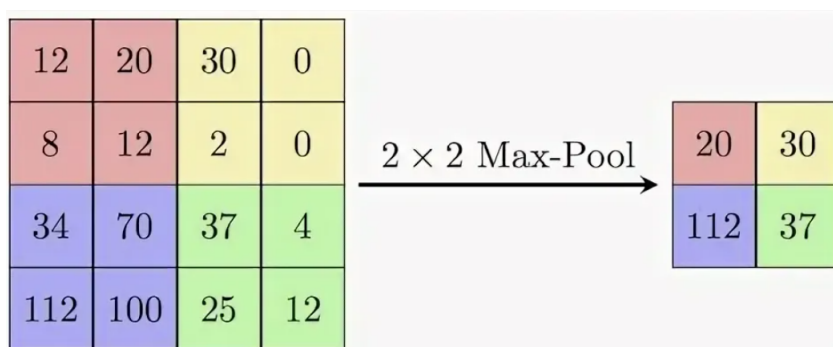


Рис. 2.3. Pooling

В декодере необходимо восстановить пространственное разрешение карты признаков. Для этого используется операция **upsampling**, которая увеличивает размер карты признаков до исходного разрешения изображения и может уменьшать число каналов.

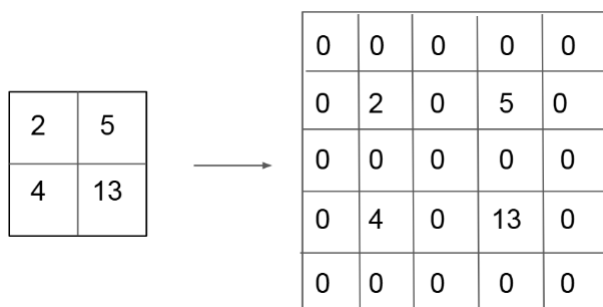


Рис. 2.4. Upsampling

Для сохранения локальной информации и точной границы объектов применяются **skip connections**, которые соединяют соответствующие слои энкодера и декодера, чтобы избежать проблему затухания градиентов. Это позволяет декодеру использовать низкоуровневые признаки из энкодера вместе с высокоуровневыми признаками, улучшая точность сегментации мелких объектов [1].

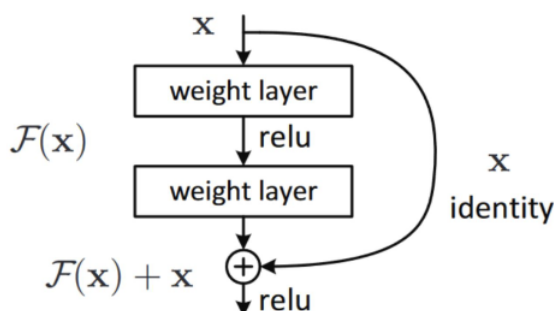


Рис. 2.5. Skip-Connection

**Энкодер** состоит из последовательности блоков: две свёртки с двумя активациями ReLU, после которых следует max-pooling. На каждом уровне уменьшается пространственное разрешение и увеличивается количество каналов, что позволяет изучать более сложные признаки.

**Декодер** выполняет upsampling карты признаков и последующие две свёртки с двумя ReLU, при этом на каждом уровне добавляется информация с соответствующего слоя энкодера через skip connections. Это позволяет восстановить пространственное разрешение и точно локализовать объекты.

**Выходной слой** состоит из свёртки  $1 \times 1$ , которая преобразует карту признаков в маску сегментации:

Таким образом, U-Net объединяет локальные признаки низкого уровня с глобальными признаками высокого уровня, обеспечивая точную попиксельную сегментацию.

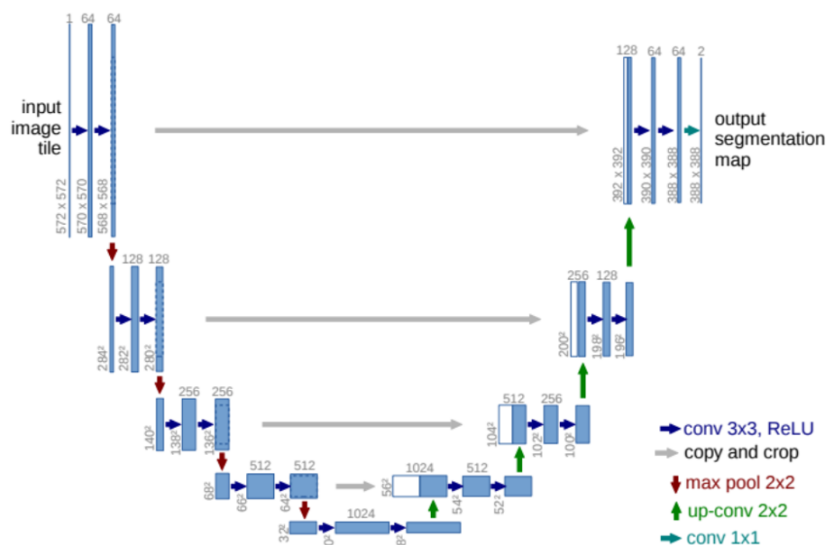


Рис. 2.6. Архитектура U-net



### 2.3. Подготовка данных

В данной работе для обучения и оценки модели семантической сегментации используется набор данных изображений рыб, содержащий 9000 пар изображений и соответствующих им масок. Исходные изображения имеют различное пространственное разрешение  $590 \times 445$  пикселей и относятся к девяти различным классам рыб.

Каждое изображение сопровождается бинарной маской, задающей принадлежность пикселей к объекту или фону. Таким образом, задача формулируется как задача бинарной семантической сегментации, где для каждого пикселя необходимо определить его принадлежность к классу объекта.

Для обеспечения корректной работы нейронной сети и возможности пакетной обработки данных все изображения и маски приводятся к фиксированному размеру  $224 \times 224$  пикселя. Выбор данного размера является компромиссом между сохранением пространственной информации и снижением вычислительной сложности модели, что особенно важно при использовании архитектуры U-Net.

Перед подачей изображений в модель выполняется их преобразование в тензорный формат и нормализация по каналам RGB. Нормализация проводится с использованием средних значений и стандартных отклонений, вычисленных для набора ImageNet.

Для масок сегментации применяется масштабирование с использованием ближайшего соседа, что позволяет сохранить дискретный характер меток и избежать появления некорректных промежуточных значений. Маски также преобразуются в тензорный формат без дополнительной нормализации.

Приведение изображений к фиксированному размеру позволяет уменьшить пространственную размерность входных данных более чем в 5 раз по сравнению с исходным разрешением. Хотя количество обучаемых параметров сети определяется архитектурой и не зависит от размера входа, такое преобразование существенно снижает объём вычислений, требования к памяти и ускоряет процесс обучения модели.

#### Деление на выборки

В глубоком обучении для корректного обучения и объективной оценки качества модели принято разделять исходный набор данных на несколько непересекающихся выборок:

- **Обучающая выборка** — используется для непосредственного обучения модели и оптимизации её параметров.
- **Валидационная выборка** — предназначена для промежуточной оценки качества модели в процессе обучения, подбора гиперпараметров и контроля переобучения.
- **Тестовая выборка** — применяется для итоговой оценки качества модели и проверки её обобщающей способности на ранее не использованных данных.

В рамках данной работы исходный датасет был разделён на обучающую, валидационную и тестовую выборки с использованием функции `random_split` библиотеки PyTorch. На первом этапе 70% данных были выделены для обучения модели, а оставшиеся 30% — для последующего разбиения. Далее данная часть была разделена на валидационную и тестовую выборки в пропорции 60% и 40%, что соответствует

18% и 12% от общего объёма данных соответственно. Такое разбиение позволяет эффективно контролировать процесс обучения и объективно оценивать её качество на отложенных данных.

#### 2.4. Метрики оценки качества модели

Для оценки качества работы модели семантической сегментации в данной работе используются две основные метрики: точность классификации по пикселям (Accuracy) и коэффициент пересечения и объединения (Intersection over Union, IoU). Обе метрики вычисляются на основе сравнения предсказанных и истинных меток пикселей.

Предсказания модели представляют собой значения вероятностей принадлежности пикселей к классу объекта. Для получения бинарной маски используется пороговая функция, согласно которой пиксель относится к классу объекта, если предсказанная вероятность превышает заданное пороговое значение.

##### IoU через множества пикселей.

Пусть  $A$  — множество пикселей, принадлежащих объекту согласно истинной разметке (истинная маска), а  $B$  — множество пикселей, отнесённых моделью к объекту (предсказанная маска). Тогда метрика Intersection over Union определяется как отношение мощности пересечения этих множеств к мощности их объединения:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}. \quad (2.4)$$

Эта формулировка наглядно показывает степень совпадения предсказанной и истинной области объекта: числитель отражает количество пикселей, корректно выделенных моделью, а знаменатель — общую область, охватывающую как истинный объект, так и его предсказание [3].

##### Матрица ошибок.

Для формального представления метрик введём элементы матрицы ошибок:

- $TP$  (True Positive) — количество пикселей, принадлежащих  $A \cap B$ ;
- $FP$  (False Positive) — количество пикселей, принадлежащих  $B \setminus A$ ;
- $FN$  (False Negative) — количество пикселей, принадлежащих  $A \setminus B$ ;
- $TN$  (True Negative) — количество пикселей, не принадлежащих ни  $A$ , ни  $B$ .

##### Accuracy.

Метрика Accuracy определяет долю правильно классифицированных пикселей относительно общего числа пикселей изображения и вычисляется по формуле:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \quad (2.5)$$

Данная метрика является простой и наглядной, однако в задачах семантической сегментации она может быть недостаточно информативной. Это связано с тем, что фон на большинстве изображений занимает значительно большую площадь по сравнению с объектом, который нужно правильно сегментировать. В результате Accuracy

может быть завышена даже при неудовлетворительном качестве сегментации объекта.

### **Intersection over Union (IoU) через матрицу ошибок.**

С использованием элементов матрицы ошибок метрика IoU может быть представлена следующим образом:

$$\text{IoU} = \frac{TP}{TP + FP + FN}. \quad (2.6)$$

В отличие от Ассигасы, метрика IoU не учитывает количество истинно отрицательных пикселей ( $TN$ ), что делает её более устойчивой к дисбалансу классов и более чувствительной к качеству сегментации именно объекта интереса [1].

### **Сравнение метрик**

Использование Ассигасы позволяет оценить общее качество попиксельной классификации, однако для задач семантической сегментации более информативной является метрика IoU. Она напрямую отражает качество совпадения предсказанной и истинной областей объекта и широко используется в качестве основной метрики в задачах сегментации изображений. В данной работе IoU рассматривается как ключевая метрика для анализа эффективности модели.

## **2.5. Функция потерь**

Для обучения модели семантической сегментации в данной работе используется функция потерь Dice loss, основанная на коэффициенте сходства Дайса. В отличие от метрик качества, функция потерь должна быть дифференцируемой, так как именно по её значению осуществляется оптимизация параметров нейронной сети с использованием алгоритма обратного распространения ошибки.

Классический коэффициент Дайса определяется через элементы матрицы ошибок следующим образом:

$$\text{Dice} = \frac{2TP}{2TP + FP + FN}. \quad (2.7)$$

Поскольку операции пороговой бинаризации не являются дифференцируемыми, при вычислении Dice loss используются непрерывные предсказания модели в виде вероятностей, получаемых с помощью сигмоидальной функции активации. Это позволяет сохранить дифференцируемость функции потерь и применять градиентные методы оптимизации [4].

В данной работе функция потерь определяется как

$$\mathcal{L}_{Dice} = 1 - \text{Dice}, \quad (2.8)$$

что соответствует минимизации несоответствия между предсказанной и истинной масками сегментации. Использование Dice loss особенно эффективно в задачах с дисбалансом классов, так как она напрямую ориентирована на качество сегментации объекта и слабо зависит от количества фоновых пикселей.

## **2.6. Результаты эксперимента**

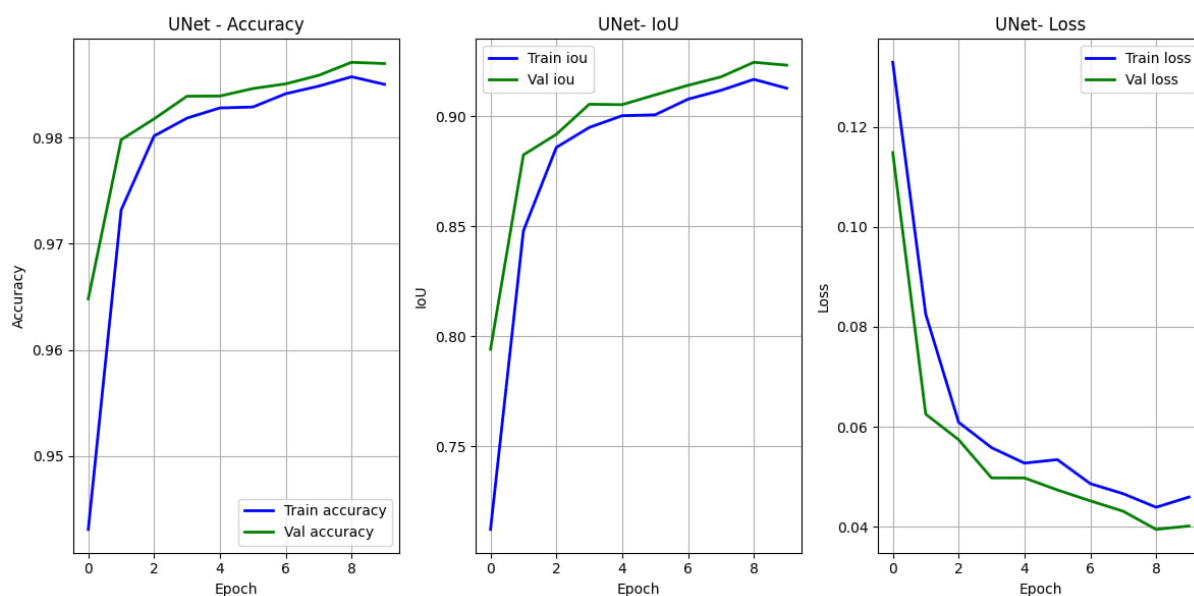
В результате обучения модели семантической сегментации на основе архитектуры U-Net была проведена оценка её качества на тестовой выборке, не участвовавшей

в процессе обучения и настройки параметров модели. Для количественной оценки использовались метрики Accuracy и Intersection over Union (IoU).

Результаты оценки качества модели на тестовой выборке у Accuracy составляет 0.9857, а у IoU — 0.9156.

Высокое значение метрики Accuracy свидетельствует о корректной попиксельной классификации изображений в целом. В то же время значение метрики IoU, превышающее 0.9, указывает на хорошее совпадение предсказанных и истинных масок объектов, что является более строгим и информативным показателем качества семантической сегментации.

Для анализа процесса обучения были построены графики изменения значения функции потерь и метрик качества на обучающей и валидационной выборках в зависимости от номера эпохи. Анализ данных графиков позволяет оценить сходимость модели и выявить возможные признаки переобучения.



**Рис. 2.7.** Метрики и функция потерь на тренировочной и валидационной выборке

Представленные графики демонстрируют устойчивое снижение значения функции потерь и рост метрик качества на протяжении обучения. Близость кривых для обучающей и валидационной выборок свидетельствует об отсутствии выраженного переобучения и хорошей обобщающей способности модели.

### 3. Заключение

В рамках данной курсовой работы была рассмотрена задача семантической сегментации изображений рыб с использованием методов глубокого обучения. Для решения поставленной задачи была выбрана и реализована архитектура сверточной нейронной сети U-Net, показавшая высокую эффективность в задачах попиксельной классификации изображений.

В ходе работы была выполнена подготовка и анализ набора данных, включающая нормализацию изображений, приведение их к фиксированному размеру и разделение на обучающую, валидационную и тестовую выборки. Была проведена реализация процесса обучения модели с использованием функции потерь Dice loss и оценка качества сегментации с помощью метрик Accuracy и Intersection over Union.

Экспериментальные результаты показали, что обученная модель демонстрирует высокое качество сегментации на тестовых данных. Значения метрик Accuracy и IoU подтверждают способность модели корректно выделять объекты и сохранять их границы даже при наличии вариаций формы и условий съёмки.

В качестве направлений дальнейшего развития работы можно рассмотреть использование методов регуляризации, таких как Dropout и Batch Normalization, для повышения устойчивости модели и снижения риска переобучения.

Полученные результаты свидетельствуют о целесообразности применения архитектуры U-Net для решения задач семантической сегментации изображений биологических объектов.

#### 4. Список литературы

1. Конушин А.С. Компьютерное зрение. — М.:МГУ, 2010. — 328 с.
2. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation [Электонный ресурс]. — 2015. — URL: <https://arxiv.org/abs/1505.04597>
3. Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, Silvio Savarese. Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression [Электонный ресурс]. — 2019. — URL: <https://arxiv.org/abs/1902.09630>
4. Chen Shen, Holger R. Roth, Hirohisa Oda, Masahiro Oda, Yuichiro Hayashi, Kazunari Misawa, Kensaku Mori. On the influence of Dice loss function in multi-class organ segmentation of abdominal CT using 3D fully convolutional networks [Электонный ресурс]. — 2018. — URL: <https://arxiv.org/abs/1801.05912>

## 5. Приложение

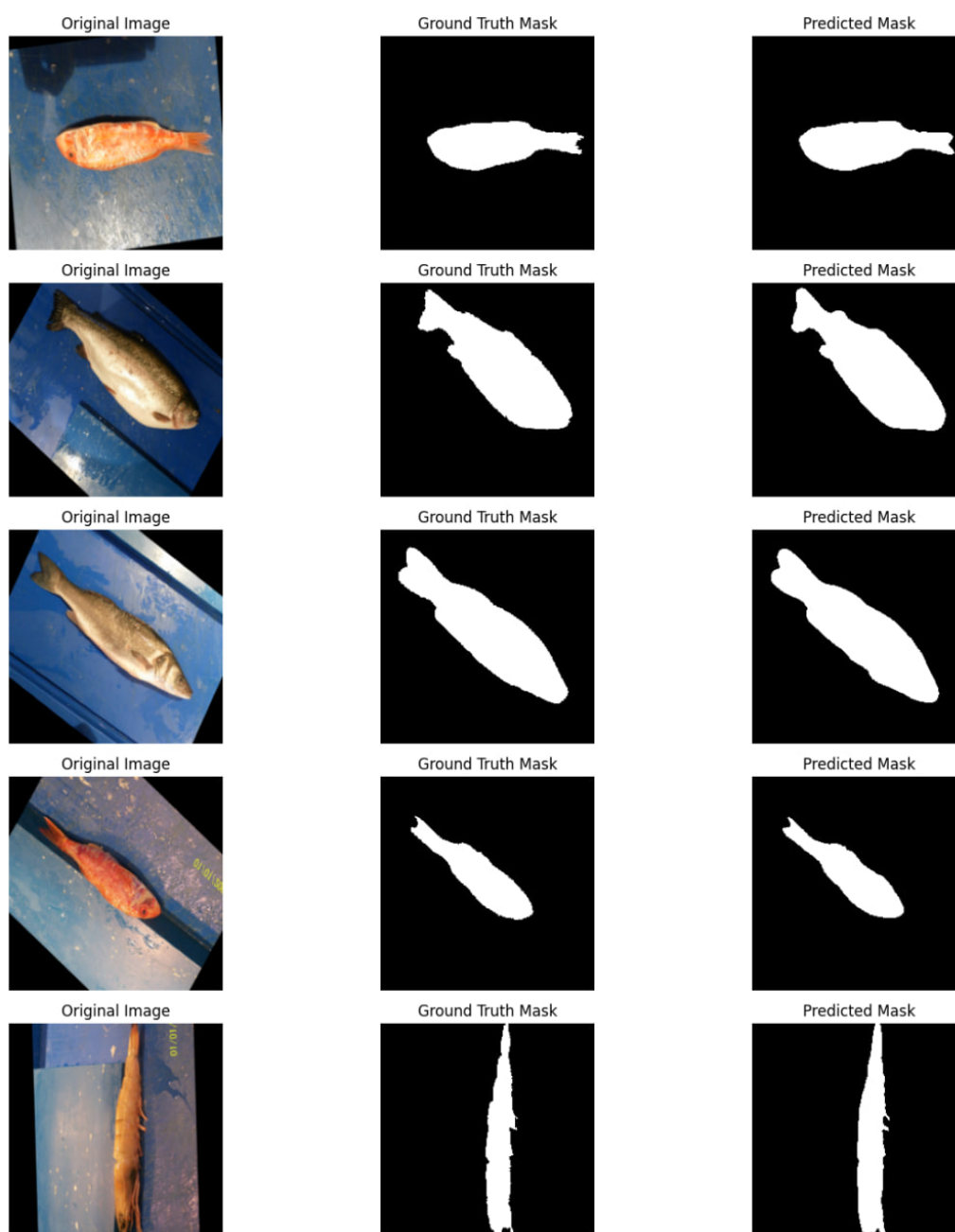


Рис. 5.1. Предсказания модели на тестовой выборке