**Пояснительная записка**

для итогового проекта на тему «Разработка готового решения для задач классификации и детекции пигментных поражений кожи» студента направления DSU-52 Литау Игоря Сергеевича.

# **Введение**

## **Немного о проблеме**

Диагностика пигментных поражений кожи является одной из актуальных задач современной медицины. Пигментные образования, такие как невусы и меланомы, представляют собой серьезную угрозу для здоровья человека. Меланома, в частности, является одной из наиболее агрессивных форм рака кожи, характеризующейся быстрым метастазированием и высокой смертностью. Согласно данным Всемирной организации здравоохранения (ВОЗ), ежегодно регистрируется около 132 тысяч новых случаев меланомы, причем заболеваемость продолжает расти, особенно в странах с высоким уровнем ультрафиолетового излучения. Ранняя диагностика и своевременное лечение значительно повышают шансы на выживание пациентов, однако сложность визуального распознавания пигментных поражений на ранних стадиях делает эту задачу крайне сложной.

Традиционные методы диагностики включают клинический осмотр, дерматоскопию и гистологическое исследование. Дерматоскопия, как один из основных инструментов, позволяет врачам детально изучить структуру пигментных образований, однако ее точность во многом зависит от опыта специалиста. Гистологический анализ, хотя и является "золотым стандартом", требует инвазивного вмешательства и не всегда применим на ранних этапах диагностики.

В последние годы все большую роль в диагностике пигментных поражений кожи играют методы компьютерного зрения и искусственного интеллекта. Эти технологии позволяют автоматизировать процесс анализа изображений кожи, повышая точность и скорость диагностики. Алгоритмы машинного обучения, обученные на больших наборах данных, способны выявлять ранние признаки меланомы и других пигментных поражений с высокой точностью, что делает их ценным инструментом для поддержки врачей.

Работа в этом направлении имеет огромное значение для улучшения качества диагностики и снижения смертности от меланомы. Разработка и внедрение систем на основе компьютерного зрения могут не только повысить доступность ранней диагностики, но и снизить нагрузку на медицинский персонал, что особенно важно в условиях растущей заболеваемости. Таким образом, интеграция современных технологий в медицинскую практику открывает новые перспективы для борьбы с пигментными поражениями кожи и улучшения здоровья населения.

## **Цели и задачи**

Исходя из вышеизложенного, целью работы является разработать готовое решение для задачи классификации и детекции пигментных поражений кожи с применением методов компьютерного зрения.

Для достижения поставленной цели необходимо решить следующие задачи:

- подобрать данные для обучения и тестирования;

- подготовить данные для обучения и тестирования;

- выбрать наилучшую модель для классификации и сегментации;

- обучить выбранные модели и оценить метрики;

- реализовать готовое решение

# **Знакомство с данными**

Для решения поставленных задач было решено выбрать данные дерматоскопических изображений, собранных в медицинских учреждениях ([ISIC](https://www.isic-archive.com/)).

## **Классификация**

Название: Skin Cancer MNIST: HAM10000.

Размер: 10,015 изображений кожных новообразований.

Классы: 7 типов кожных заболеваний:

* akiec — Actinic keratoses (актинический кератоз и плоскоклеточный рак);
* bcc — Basal cell carcinoma (базальноклеточная карцинома);
* bkl — Benign keratosis-like lesions (доброкачественные кератозы);
* df — Dermatofibroma (дерматофиброма);
* mel — Melanoma (меланома);
* nv — Melanocytic nevi (меланоцитарный невус);
* vasc — Vascular lesions (сосудистые поражения);

Характеристики изображений:

* Разрешение: 450x600 пикселей;
* Формат: JPEG.

Преимущества датасета:

* Большой объем данных (10,015 изображений);
* Разнообразие классов (7 типов заболеваний);
* Высокое качество изображений.

Проблемы датасета:

* Дисбаланс классов;
* Необходимость предобработки данных (нормализация, аугментация).

Самая большая проблема датасета это дисбаланс. Класс nv (Melanocytic nevi) составляет 67% от всех данных.

Для решения этой проблемы использовали 3 подхода:

* взвешивание классов (class weighting);
* вычисления весов для сэмплера и последующей балансировке батчей
* (WeightedRandomSampler);
* oversampling (за счет дублирования изображений минорных классов) и последующие transforming и augmenting (для предупреждения переобучения модели);
* парсинг веб-источников для расширения датасета.

Взвешивание классов показало слабые значения метрик на train, val и test.

Вычисления весов для сэмплера и последующая балансировка батчей так же не дали желаемого результата. Данный подход показал хорошие значения на train, но плохие на val и test что говорит о переобучении.

Искусственное увеличение за счет oversampling показало хорошие значения метрик на train, но плохие на val и test что говорит о том, что модель переобучилась.

Наилучшим решением оказался parsing. В качестве данных использовались дерматоскопических изображений из коллекции ISIC Archive. Увы, полностью сбалансировать датасет не получилось, но некоторые классы удалось значительно увеличить.

Тем не менее, oversampling это очень сильный инструмент при работе с плохо сбалансированными данными. В нашем случае, мы пытаемся увеличить минорные классы до размера самого большого класса в датасете и множители для классов равны: bkl (Benign keratosis-like lesions) - 6, df' (Dermatofibroma) – 62(!), mel (Melanoma) - 6, vasc (Vascular lesions) – 44(!), bcc (Basal cell carcinoma) - 13, akiec (Actinic keratoses) - 21. Некоторые множители просто огромные и переобучение моделей уже не кажется удивительным.

Для решения этой проблемы был введен поправочные коэффициент, который уменьшал множители. Это позволило увеличить минорные классы и избежать переобучения.

В итоге, на вход модели поступали данные, полученные после парсинга и oversampling с поправочным коэффициентом.

**Детекция**

Название: HAM10000 Lesion Segmentations.

* Размер: 10,015 segmentation masks.
* Классы: нет

Характеристики изображений:

* Разрешение: 450x600 пикселей.
* Формат: PNG.

Особых преимуществ и проблем с данными нет, т.к. это специально подготовленные маски для датасета Skin Cancer MNIST: HAM10000.

# **Решение задачи мультиклассовой классификации**

## **Выбор модели классификации**

Сверточные нейронные сети (CNN) являются одним из ключевых инструментов в задачах компьютерного зрения, включая классификацию изображений. Они успешно применяются благодаря своей способности автоматически извлекать иерархические признаки из данных, что делает их особенно эффективными для работы с изображениями.

Для решения нашей задачи было принято решение протестировать три CNN:

* **TF-EfficientNetV2** (2021) — это улучшенная версия EfficientNet, разработанная Google для баланса между скоростью и точностью. Модель сочетает MBConv и Fused-MBConv блоки, что ускоряет обучение и снижает вычислительные затраты. Эффективно масштабируется (S, M, L версии), достигая до 83% Top-1 Accuracy на ImageNet. Идеальна для задач классификации и трансферного обучения, но требует тонкой настройки гиперпараметров.
* **MobileViTv2** (2022) — гибридная архитектура, объединяющая MobileNet и Transformer. Оптимизирована для мобильных и edge-устройств: малый размер (от 5M параметров) и низкое энергопотребление. Точность скромнее (до 78% Top-1), но модель идеальна для встраиваемых систем, IoT и приложений реального времени. Главный плюс — работа на слабом железе без серьёзных потерь в качестве.

Еще одной особенностью этих CNN является их небольшой размер, что важно, т.к. в дальнейшем планируется развернуть модель на сервере и реализовать телеграмм-бота.

## **Обучение модели классификации**

Для обучения описанных выше моделей данные разделяются на обучающий (train, 80%), валидационный (val, 10%) и тестовый (test, 10%) наборы. К данным применяются аугментации: изменение размера до 256x256 пикселей, случайные отражения по горизонтали и вертикали, а также нормализация. Данные загружаются батчами по 32 изображения.

Модель CNN инициализируется с предобученными весами, после чего заменяется выходной слой для соответствия количеству классов в задаче. Модель переносится на GPU (если доступно). В качестве функции потерь используется CrossEntropyLoss, а для оптимизации — Adam с learning rate 0.001.

Обучение проводится в течение 5 эпох. На каждой эпохе модель обучается на тренировочных данных, после чего оценивается на валидационном и тестовом наборах. Метрики (потери и точность для train и val) записываются в CSV-файл для последующего анализа. После обучения модель тестируется на тестовом наборе, и строится матрица ошибок для оценки качества классификации.

В завершение визуализируются графики потерь и точности для train и val, что позволяет оценить динамику обучения и выявить переобучение. Этот процесс обеспечивает эффективное обучение модели с контролем качества на каждом этапе.

## **Оценка метрик**

Как было сказано ранее, для обучения мы использовали данные, полученные при помощи парсинга и прошедшие через oversampling с попровачным коэффициентом. Сравнивались 5 варианта:

* вариант 1 - parsing без oversampling;
* вариант 2 - parsing + oversampling (без поправочного коэффициента);
* вариант 3 - parsing + oversampling (поправочный коэффициент равен 2);
* вариант 4 - parsing + oversampling (поправочный коэффициент равен 5);
* вариант 5 - parsing + oversampling (поправочный коэффициент равен 10).

Результат оценивался на основе метрик accuracy, loss и F1-score. Наилучшие показатели продемонстрировали модели MobileViTv2, существенно превосходящие EfficientNetV2 по всем ключевым параметрам.

Среди всех протестированных моделей явным лидером стала MobileViTv2 (вариант 5), показавшая максимальную точность (accuracy = 0.890) при минимальной ошибке (loss = 0.302). Однако анализ по классам выявил ее слабое место - низкий F1-score (0.54) для класса 0, который представлен всего 29 примерами. Это делает данную модель менее предпочтительной для задач, где важна корректная классификация редких категорий.

Наиболее сбалансированные результаты показала MobileViTv2 (вариант 3). При незначительно меньшей точности (0.887) и чуть большем loss (0.322), она демонстрирует лучший средний F1-score по всем классам (macro avg = 0.85). Особенно важно, что эта модель лучше других справляется с классификацией проблемного класса 0 (F1 = 0.67), что делает ее оптимальным выбором для работы с несбалансированными данными. Также она показывает стабильно высокие результаты для всех остальных классов, без явных провалов.

Модель MobileViTv2 (вариант 1) занимает промежуточное положение с accuracy 0.888 и loss 0.320. Ее показатели F1-score для разных классов варьируются в диапазоне 0.60-0.95, демонстрируя надежную, но не выдающуюся работу. Эта модель может рассматриваться как "золотая середина", когда не требуется экстремальных значений ни по одному из параметров.

Среди моделей EfficientNetV2 лучшие результаты у EfficientNetV2 (вариант 4) (accuracy 0.872, loss 0.355), однако они существенно уступают MobileViTv2 по всем показателям. Основная проблема этих моделей - низкая эффективность работы с редкими классами, особенно с классом 0, где F1-score не превышает 0.63 даже у лучшей представительницы семейства.

Графики потерь (Loss) и графики точности (Accuracy) схожи у обеих моделей без каких-либо аномалий.

**Вывод:**

Наилучшей моделью для данной задачи является MobileViTv2 (вариант 3), так как она обеспечивает наивысшую точность и стабильность по всем ключевым метрикам.

# **Решение задачи сегментации**

## **Выбор модели сегментации**

Сверточные нейронные сети (CNN) играют ключевую роль в задачах сегментации объектов на изображениях. Сегментация — это процесс разделения изображения на области, соответствующие различным объектам или частям объектов. CNN эффективно справляются с этой задачей благодаря своей способности автоматически извлекать иерархические признаки из изображений. В отличие от классификации, где CNN определяет класс всего изображения, в сегментации сеть должна присвоить метку каждому пикселю изображения.

Для решения нашей задачи было принято решение протестировать две CNN:

* **U-Net**: Эта архитектура использует симметричную структуру с энкодером и декодером, соединенными skip-connections. Эти соединения передают информацию из энкодера в декодер, что помогает сохранить детали изображения и улучшить точность сегментации на границах объектов. U-Net особенно эффективна в задачах биомедицинской сегментации, где важна высокая точность;
* **DeepLabV3**: Эта архитектура использует атрокусные свертки (atrous convolutions) и пирамидальные pooling-слои (ASPP), чтобы захватывать контекст на разных масштабах. Это позволяет модели лучше справляться с объектами разного размера и сложными сценами. DeepLabV3 часто используется в задачах сегментации сцен, таких как автономное вождение, где важно учитывать контекст.

Каждая из этих архитектур находит свое применение в различных задачах. U-Net лучше подходит для задач, где важна точность на границах объектов (например, медицинские изображения), а DeepLabV3 — для задач, где требуется учет контекста и объектов разного масштаба (например, сегментация сцен).

## **Обучение модели сегментации**

Половину от исходного набора данных делили на обучающий (train, 80%), валидационный (val, 5%) и тестовый (test, 15%) наборы. Это делалось для экономии времени и вычислительных ресурсов. К данным применялись аугментации и преобразования: изображения изменялись до размера 256x256 пикселей и нормализовались для улучшения сходимости модели.

Модели U-Net или DeepLabV3 инициализирвали с заданными параметрами. В качестве функции потерь использовали BCEWithLogitsLoss (бинарная кросс-энтропия с логитами), а IoU (Intersection over Union) и Dice Coefficient для оценки точности моделей. Эти функции позволяли оценивать, как точность классификации пикселей, так и качество сегментации на уровне объектов. Оптимизатор Adam с learning rate 0.001 и weight decay 0.00001.

Обучение проводится в течение 5 эпох. На каждой эпохе модель обучается на батчах размером 16 изображений. После каждой эпохи модель оценивается на валидационном наборе, чтобы отслеживать переобучение. Метрики, такие как Train Loss, Train IoU Accuracy, Train Dice Accuracy, Val Loss, Val IoU Accuracy и Val Dice Accuracy, записываются в CSV-файл для последующего анализа.

Для визуализации результатов накладывались оригинальная маска и предсказанная маска на исходное изображение, что позволяло наглядно оценить качество сегментации. Также строялись графики потерь и точности (IoU и Dice) для обучающего и валидационного наборов, чтобы отслеживать динамику обучения.

## **Оценка метрик**

**DeepLabV3** демонстрирует следующие результаты: время обучения на 5 эпох составляет 43.61 минуты, значение функции потерь (BCEWithLogitsLoss) — 0.175, метрика IoU — 0.791, а Dice Coefficient — 0.882. Графики показывают, что потери (train\_loss) снижаются, но не так быстро, как у U-Net, а значения IoU и Dice на валидации (val\_jou и val\_dice) достигают уровня около 0.6–0.8. Это указывает на хорошую производительность модели, но с более длительным временем обучения и меньшей точностью.

**U-Net** показывает лучшие результаты: время обучения — 31.75 минуты, значение функции потерь — 0.152, IoU — 0.892, а Dice Coefficient — 0.942. Графики подтверждают, что потери (train\_loss) снижаются быстрее, а значения IoU и Dice на валидации достигают более высоких значений (около 0.8–0.9). Это свидетельствует о более высокой точности и эффективности модели по сравнению с DeepLabV3.

**Вывод:** U-Net является лучшей моделью для задачи сегментации изображений, так как она обеспечивает более высокую точность (IoU и Dice Coefficient), меньшее значение функции потерь и меньшее время обучения. DeepLabV3, хотя и демонстрирует достойные результаты, уступает по всем ключевым метрикам. Таким образом, U-Net можно рекомендовать для использования в задачах, где важны точность и скорость обучения.

В завершении, модель **U-Net** обучалась на всех данных в течении 5 эпох (**U-Net full**). Метрики финального обучения составили:

* Test Loss - 0.035;
* IoU (Intersection over Union) - 0.952;
* Dice Coefficient - 0.975;
* Train Time - 77.58 мин.

# **Реализация готового решения**

## **Комплексный подход**

При дерматоскопии, с применением Computer Vision, не всегда достаточно только классифицировать новообразование, иногда область повреждения сложно точно идентифицировать. Именно поэтому в данной работе был применён комплектный подход с объединением CNN для классификации и сегментации изображений. Это может помочь как специалистам, так и обычным людям, которые следят за состоянием своей кожи и контролируют появление новообразований. Конечно же данный «инструмент» не обладает 100% точностью и не способен ставить диагноз, но обратить внимание пользователя на подозрительные участки кожи и тем самым стимулировать визит к специалисту – уже большой вклад в сохранение здоровья населения.

Для обобщения этих методов был реализован класс **FinalProject**. Который на вход принимает изображение, а на выходе возвращает класс изображения, его описание (характеристики, риски, лечение и профилактика), а также исходное изображение с выделенной областью.

# **Полезные ссылки**

Статья [The International Skin Imaging Collaboration](https://www.isic-archive.com/)

Статья [Densely Connected Convolutional Networks](https://arxiv.org/abs/1608.06993)

Статья [Deep Residual Learning for Image Recognition](https://arxiv.org/abs/1512.03385)

Статья [Very Deep Convolutional Networks for Large-Scale Image Recognition](https://arxiv.org/abs/1409.1556)

Статья [U-Net: Convolutional Networks for Biomedical Image Segmentation](https://arxiv.org/abs/1505.04597)

Статья [Rethinking Atrous Convolution for Semantic Image Segmentation](https://arxiv.org/abs/1706.05587)

[The HAM10000 dataset (Harvord)](https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T)

[Skin Cancer MNIST: HAM10000 (Kaggle)](https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000/data)

[HAM10000 Lesion Segmentations (Kaggle)](https://www.kaggle.com/datasets/tschandl/ham10000-lesion-segmentations)