**Аннотация**

В данной работе рассмотрено использование нейронных сетей ResNeXt-50, ResNet-152 и EfficientNet-B0 в качестве экстрактора признаков модели, решающей задачу семантической сегментации человеческих силуэтов на фотографиях. Использование нейросетей семейства *ResNeXt* представляет особый интерес, т.к. позволяет уменьшить количество параметров модели по сравнению с ResNet и при этом не ухудшить качество решения. Уменьшение количества параметров модели мотивировано желанием использовать полученную модель в пользовательских приложениях на мобильных устройствах, где вопрос эффективности по времени работы и объему занимаемой памяти особенно актуален. В работе приведены результаты численных экспериментов , которые помогают оценить преимущества использования в качестве экстрактора признаков нейронной сети семейства *ResNeXt* в сравнении с другими архитектурами, используемыми в задачах компьютерного зрения. Среди всех рассматренных в данной работе моделей энкодеров нейронная сеть *ResNeXt-50* показала неизменно лучшее качество сегментации (по метрике Intersection over Union (IoU)) в задачах с датасетами *APSIS* и *CamVid*. С другой стороны она превзошла по скорости работы более легковесную модель EfficientNet-B0. В эксперименте с датасетом APSIS *ResNeXt-50* позволила с высокой точностью ($IoU \geq 0.96$) сегментировать портреты людей. В эксперименте с датасетом CamVid использование архитектуры с энкодером *ResNeXt-50* помогло получить прирост в качестве сегментации (примерно 6%) по сравнению с более тяжеловесной моделью ResNet-152.

**Содержание**

## 1. Введение

### *1.1. Описание проблемы*

Сегментация изображений — это процесс присвоения таких меток каждому пикселю изображения, что пиксели с одинаковыми метками имеют общие визуальные характеристики. Результатом сегментации изображения является множество сегментов, которые вместе покрывают всё изображение, или множество контуров, выделенных из изображения. Некоторыми практическими применениями сегментации изображений являются: выделение объектов на спутниковых снимках, распознавание лиц, обнаружение опухолей и других патологий в медицинских изображениях, распознавание отпечатков пальцев и т.д.

Стоит заметить, что у портретной сегментации есть свои уникальные требования. Разработка легкого и надежного алгоритма сегментации портрета - важная задача для широкого спектра пользовательских приложений для обработки фотографий лиц. Поскольку портретная сегментация выполняется во многих приложениях в реальном времени, то для нее требуются чрезвычайно легкие с точки зрения количества параметров модели.

Внутри архитектуры модели для сегментации изображений можно выделить две основные части - это энкодер и декодер. Энкодер (или подругому - экстрактор признаков) сжимает входные данные для представления их в скрытом пространстве малой размерности. Декодер же осуществляет восстанавление из этого представления данных, которые будут являться выходом всей архитектуры.

В данной работе будет проанализирован нейросетевой метод сегментации портретов людей, в котором в качестве энкодера будет выбрана модель *ResNeXt-50* ([1]), а в качестве декодера - модель U-Net ( [2]). Основная цель работы - показать, что использование в качестве энкодера модели семейства *ResNeXt* позволяет получить достаточно высокое качество сегментации человеческих портретов по сравнению с более легковесными сетями типа EfficientNet-B0 ([3]) и при этом уменьшить количество параметров модели по сравнению с аналогами из семейства ResNet.
Уменьшение количества параметров модели мотивировано желанием использовать данную архитектуру в мобильных устройствах, где вопрос эффективности по времени работы и объему занимаемой памяти особен-

но актуален. Повышение эффективности нейронной сети не только улучшает пользовательский опыт за счет более высокой точности и меньшей задержки, но также помогает продлить срок службы батареи за счет снижения энергопотребления. Будет подобран размеченный датасет с человеческими портретами, на котором будет производиться обучение нейронных сетей и их сравнение. Будут проведены численные эксперименты, цель которых - оценить качество сегментации с помощью архитектуры, в которой в качестве энкодера выступает модель *ResNeXt-50* , а также сравнить результат, получаемый тремя различными способами ( с энкодерами *ResNeXt-50* , ResNet-152, EfficientNet-B0 ) по метрике IoU.

### *1.2. Обзор литературы*

Сложность задачи сегментации заключается в том, что модель должна одновременно решать две противоположные задачи: 1) обработка зависимостей, присутствующих на всем наборе изображений, и 2) сохранение подробной локальной информации, уникальной для каждого конкретного изображения. Это приводит к тому, что модели с небольшим числом параметров зачастую демонстрируют невысокое качество сегментации даже на простых входных изображениях . Интересные результаты были получены в работе [4], в которой авторы смогли добиться значительного уменьшения количества параметров модели, использовавшейся для сегментации человеческих портретов. Авторы представили новую чрезвычайно легкую модель сегментации портретов SINet, содержащую декодер блокировки информации и модули пространственного сжатия. Предложенный ими метод позволил сократить количество параметры от 2,1 млн. до 86,9 тыс. (снижение примерно на $95,9\%$), при уменьшении точности менее 1% по сравнению с актуальными методами сегментации портретов. Кроме этого они показывают, что модель успешно работает на мобильном телефоне, при этом частота обновления кадров оказывается не менее 100 FPS.

В работе [5] впервые была предложена идея построить полностью сверточные сети (*англ.* Fully Convolutional Networks (FCN)) на основе таких моделей, как AlexNet ([6]), VGG ([7]), GoogLeNet([8]) с помощью замены полносвязных слоев сверточными и добавления деконволюционного слоя. При этом модели могли обрабатывать изображения произвольного размера, что позволило увеличить размеры обучающих выборок посредством разделения каждого изображения на фрагменты и их исполь-

зования при обучении. В FCN повысилась эффективность обучения по сравнению с актуальными на тот момент моделями для семантической сегментации, при этом достиагалось неплохое качество сегментации ( IoU = 62% в задаче PASCAL VOC).

В модели SegNet, предложенной в статье [9], была реализована архитектура, состоящая из энкодера и декодера. Энкодер состоял из последовательности сверточных и пулинг слоев, которые выполняли отображение исходного изображения в более маломерное признаковое пространство. Декодер же осуществлял разжатие изображения, используя сверточные и пулинг слои. Преимуществом такого решения является то, что в отличие от FCN оно производит разжатие изображения до исходного размера постепенно и делает это, используя информацию о том, какие пиксели использовались при пуллинге в блоке энкодера.

Модель U-Net ([10]), созданная на основе традиционной сверточной нейронной сети, была разработана и впервые применена в 2015 году для обработки биомедицинских изображений. В настоящее время U-Net является одной из наиболее популярных нейронных сетей для сегментации изображений, а статья [10] имеет более 27 тысяч цитирований на июнь 2021 года. Архитектура сети представляет собой полносвязную свёрточную сеть, модифицированную таким образом, чтобы она могла работать с меньшим количеством примеров (обучающих образов) и делала более точную сегментацию. Также в U-Net используется прием под названием пропускное соединение (*англ. scip connection*), который используется для задач, в которых выход имеет такое же пространственное измерение, что и входные данные (сегментация изображения, оценка оптического потока, прогнозирование видео и т.д.). Как оказалось, симметричные соединения с длинными пропусками работают невероятно эффективно в задачах плотного прогнозирования (сегментация медицинских изображений).

Обычно нейронная сеть, инициализированная весами из сети, предварительно обученной на большом наборе данных, таком как ImageNet, демонстрируют лучшее качество работы , чем те, которые обучены с нуля на небольшом наборе данных. Это наблюдение привело авторов работы [11] к созданию модели TernausNet. В ее основе лежит идея замены сверточной части U-Net ([10]) энкодером VGG11 ([12]), который был предварительно обучен на датасете ImageNet. Как показывают авторы в

работе [11], в результате замены энкодера предобученной моделью произошел значительный прирост (около 10% по метрике IoU) качества сегментации в задаче Kaggle: Carvana Image Masking Challenge. Результаты экспериментов, описанные в [11], явно указывают на то, что TernausNet в процессе обучения гораздо быстрее сходится к стабильному значению метрики IoU по сравнению с не прошедшей предварительное обучение сетью.

Создание нейронных сетей ResNet ([13]) внесло существенный вклад в решение проблемы обучения глубоких нейронных сетей. Во многих задачах глубокие нейросети показывали невысокие результаты как при обучении, так и при тестировании. Создатели ResNet предположили, что проблема заключается в оптимизации — более глубокие модели гораздо хуже поддаются настройке. Тогда они решили не просто добавлять дополнительные слои для изучения отображения нужной функции напрямую, а использовать остаточные блоки, которые помогают правильным образом подбирать это отображение.Так ResNet стала первой остаточной нейронной сетью.

Нейронные сети семейства *ResNeXt* были впервые предложены в работе [1] для решения задачи классификации объектов на изображении. Авторы, представившие архитектуру *ResNeXt* , использовали ее на соревновании по классификации ILSVRC 2016, в котором заняли второе место. В статье [1] авторы дополнительно оценивают *ResNeXt* на более крупных наборах данных ImageNet-5K и COCO. Как показывают результаты их экспериментов, нейросети семейства *ResNeXt* превосходят ResNet-101/152, ResNet-200, Inception-v3 и Inception-ResNet-v2 по классификации изображений из датасета ImageNet. В частности, 101-слойная модель *ResNeXt* может обеспечить лучшую точность, чем ResNet-200, при этом она имеет в два раза меньшее количество параметров. В проведенных экспериментах *ResNeXt* демонстрирует неизменно лучшую точность, чем его аналог ResNet. Также авторами высказывается предположение, что *ResNeXt* может быть хорошо обобщен для других задач визуального распознавания.

В 2019 году группа разработчиков Google опубликовала работу ([3]). Основная идея публикации заключалась в стратегическом масштабировании глубоких нейронных сетей посредством нового семейства нейронных сетей EfficientNet. Метод масштабирования (*англ. compound scaling*

*method*), представленный в статье, предлагает вместо масштабирования только одного из атрибутов модели (глубины, ширины и разрешения входного изображения) использовать стратегическое масштабирование всех трех из них вместе, что в свою очередь, помогает повысить качество работы нейронной сети.

## 2. Постановка задачи

В данной работе исследуется проблема выбора энкодера для решения задачи сегментации человеческих фигур на изображениях, который бы позволил уменьшить количество параметров в используемой архитектуре и одновременно сохранил высокое качество сегментации. Кроме того, важным фактором является время вывода модели, которое также необходимо учитывать при выборе архитектуры. Требуется проанализировать различные модели энкодеров в работе на реальных изображениях людей и подобрать наиболее оптимальный по сложности и качеству решения задачи сегментации энкодер.

## 3. Методы

В данном разделе будут описаны архитектуры нейронных сетей, которые выбраны для сравнения в производительности и качестве решения задачи сегментации человеческих силуэтов на изображении. Будут рассмотрены 3 различные архитектуры на базе нейронных сетей *ResNeXt-50* , ResNet-152, EfficientNet-B0. При этом будет позаимствована идея, предложенная создателями TernausNet в работе [11], и будут использованы предобученные на датасете ImageNet модели энкодеров. Во всех трех архитектурах в качестве декодера будет использована нейронная сеть U-Net. Выбор моделей ResNet-152, EfficientNet-B0 для сравнения с *ResNeXt-50* обоснован их широкой применимостью в задачах компьютерного зрения ([14], [15], [16], [17], [18], [19]). При этом они являются представителями тяжеловесных и легковесных моделей соответственно.

### *3.1. Архитектура модели с энкодером ResNeXt-50*

Нейронная сеть *ResNeXt* представляет собой совокупность параллельных ветвей, в каждой из которых осуществляется одинаковая последо-

вательность преобразований над входным вектором. Такой дизайн архитектуры позволяет уменьшить число гиперпараметров модели. В частности, возникает новое измерение, которое авторы архитектуры ([1]) называют мощностью (*англ. cardinality* ), которое определяет размер набора преобразований и является важным фактором в дополнение к глубине и ширине. Общее количество параметров модели *ResNeXt-50* примерно равно $22 \cdot 10^6$.

| | ***ResNeXt-50*** $(\mathbf{32 \times 4})$ |
|---|---|
| Вход | $512 \times 512 \times 1$ |
| Слой 1 | Conv2D $(7 \times 7, 64, \text{stride } 2)$ |
| Слой 2 | $3 \times 3$ max pool, stride 2 |
| Слой 3 | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128, C = 32 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| Слой 4 | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256, C = 32 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ |
| Слой 5 | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512, C = 32 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$ |
| Слой 6 | $\begin{bmatrix} 1 \times 1, 1024 \\ 3 \times 3, 1024, C = 32 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ |
| Выход | $1 \times 1$ |

Таблица 1: Архитектура энкодера для сегментации с *ResNeXt-50*

Рассмотрим архитектуру декодера, который в нашем случае представлен нейронной сетью U-Net (см. Рис. 1).

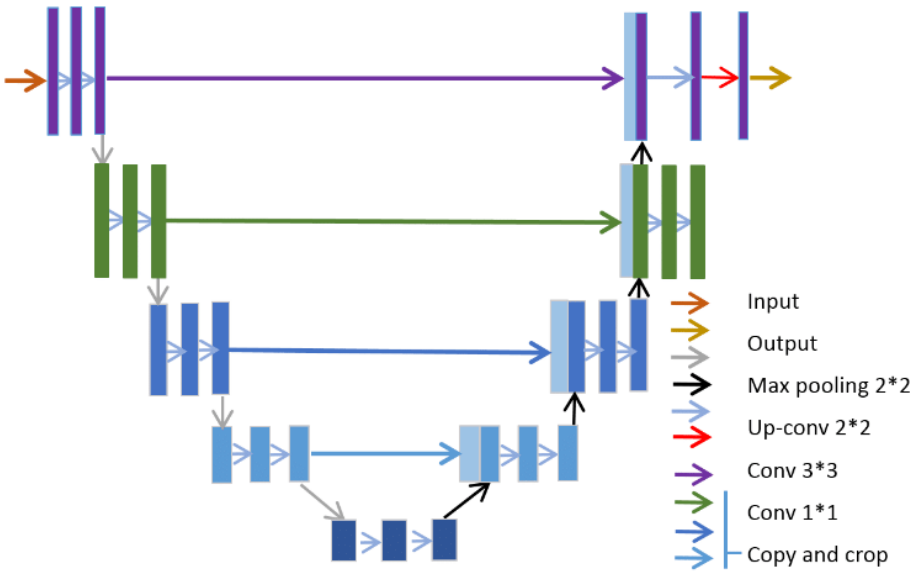


Рис. 1: Архитектура декодера для сегментации

Сеть содержит сверточную (слева) и разверточную (справа) части, поэтому архитектура похожа на букву U, что и отражено в названии.

Энкодер похож на обычную свёрточную сеть, он содержит два подряд свёрточных слоя $3 \times 3$, после которых идет слой ReLU и пулинг с функцией максимума $2 \times 2$ с шагом 2.

Каждый шаг декодера содержит слой, обратный пулингу, который расширяет карту признаков. После этого следует свертка $2 \times 2$, которая уменьшает количество каналов признаков. Далее идет конкатенация с соответствующим образом обрезанной картой признаков из сжимающего пути. Этот прием позволяет учитывать более низкоуровневое признаковое представление входного изображения при его декодировании. Далее следуют две свертки $3 \times 3$, после каждой из которой идет ReLU. Обрезка нужна из-за того, что происходит потеря пограничных пикселей в каждой свёртке. На последнем слое свертка $1 \times 1$ используется для приведения каждого 64-компонентного вектора признаков до требуемого

количества классов. Всего сеть имеет 23 свёрточных слоя.

### *3.2. Архитектура модели с энкодером ResNet-152*

Основной элемент ResNet — остаточный блок (*англ. Residual block*) с shortcut-соединением, через которое данные проходят без изменений. Остаточный блок (см. Рис.2) представляет собой несколько свёрточных слоёв с активациями, которые преобразуют входной сигнал $x$ в $F(x)$. Shortcut-соединение — это тождественное преобразование $x \to x$

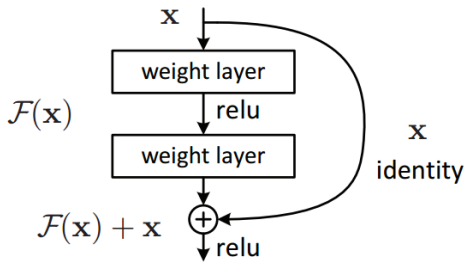


Рис. 2: Остаточный блок в нейронной сети ResNet

| | **ResNet-152** |
|---|---|
| Вход | $512 \times 512 \times 1$ |
| Слой 1 | Conv2D $(7 \times 7, 64, \text{stride } 2)$ |
| Слой 2 | $3 \times 3$ max pool, stride 2 |
| Слой 3 | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| Слой 4 | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$ |
| Слой 5 | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$ |
| Слой 6 | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ |
| Выход | $1 \times 1$ |

Таблица 2: Архитектура энкодера для сегментации с ResNet-152

### *3.3. Архитектура модели с энкодером EfficientNet-B0*

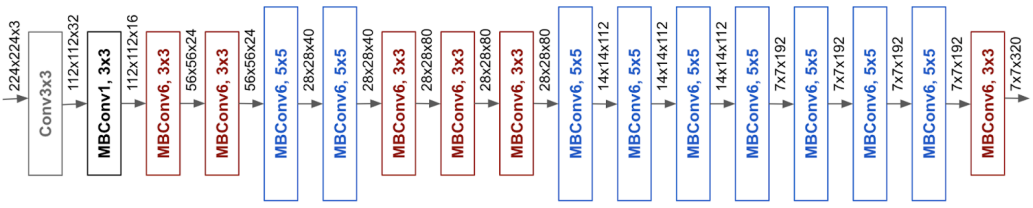


Рис. 3: Архитектура энкодера для сегментации с EfficientNet-B0

| Энкодер | Количество параметров (млн) |
|---|---|
| ResNeXt-50 | 22 |
| ResNet-152 | 60 |
| EfficientNet-B0 | 4 |

Таблица 3: Сравнение количества параметров для используемых энкодеров

## 4. Датасеты

### *4.1. APSIS*

Обучение моделей производилось на данных, предложенных в работе [20]. Авторы подобрали 1800 портретных изображений с веб-сайта *flickr.com* и вручную разметили их с помощью быстрого редактирования в Photoshop. Затем они запускали детектор лиц для каждого изображения и автоматически масштабировали и кадрировали изображение до разрешения $600 \times 800$ пикселей в соответствии с ограничивающей рамкой обнаружения лица. Датасет содержит как портретные фотографии людей, так и автопортреты, снятые на мобильные фронтальные камеры. Пример фотографии и маски из обучающей выборки датасета приведены на Рис. 3. Таким образом в датасете представлены типичные случаи, которые требуется научиться обрабатывать. Изображения из датасета очень хорошо подходят для обучения модели, т.к. позволяют модели после случайной начальной инициализации научиться обрабатывать характерные фрагменты человеческих фотографий.

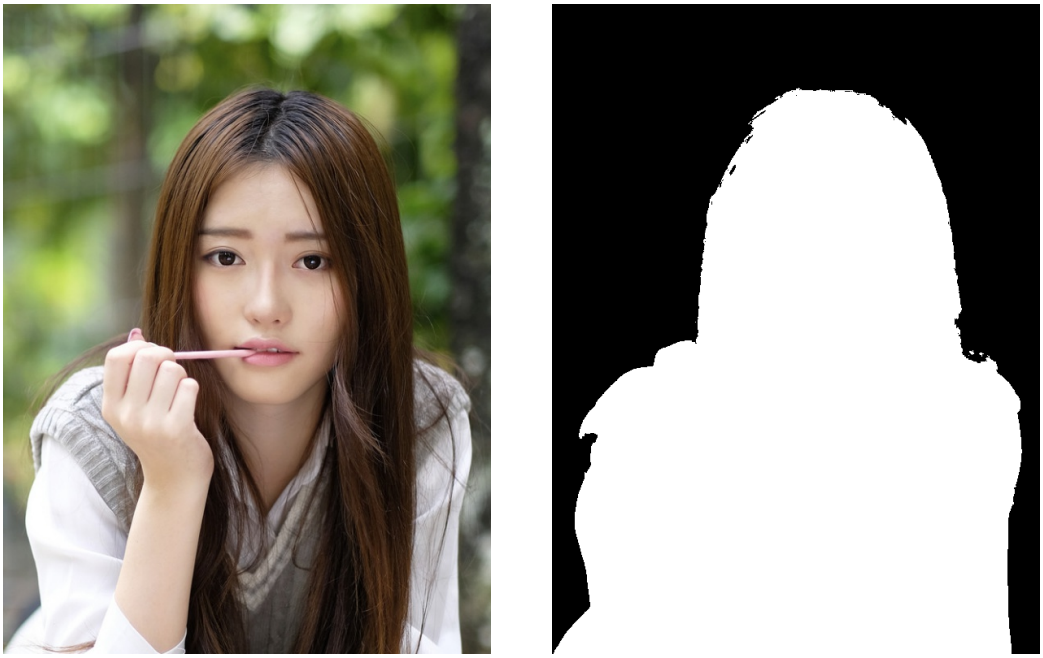

Рис. 4: Пример изображения и маски из обучающей выборки датасета *APSIS*

### *4.2. CamVid*

Кембриджский датасет размеченных видео (*англ. Cambridge-Driving Labeled Video Database (CamVid)*) - это известный датасет с более чем 700 размеченными фотографиями с разрешением $320 \times 480$, на которых представлены 32 различных класса объектов, в том числе и люди. Попиксельная семантическая сегментация более 700 изображений была произведена вручную, а затем проверена и подтверждена несколькими специалистами на предмет точности. Данный датасет изначально был предназначен для обучения моделей, использующихся в беспилотных транспортных средств, однако люди распределены на изображениях таким образом, что в контексте решаемой задачи он представляет интерес, т.к. для успешной сегментации человеческих фигур на изображениях из датасета используемая нейросетевая модель должна обладать большой обобщающей способностью. Обучение моделей на таком датасете позволяет оценить их применимость в ситуации, когда на фотографии одновременно сразу несколько людей и нет строгой упорядоченности в их взаимном расположении.

Рис. 5: Пример изображения и маски из обучающей выборки датасета *CamVid*

## 5. Эксперименты

Эксперименты были реализованы на языке программирования Python с помощью фреймворка PyTorch [21]. Все эксперименты, в том числе дообучение моделей на датасетах APSIS и CamVid, были проведены с использованием видеокарты NVIDIA GeForce RTX -2080 Ti.

Для обучения моделей используется оптимизатор Adam ([22]), начальное значение шага обучения $\alpha = 10^{-4}$, каждые 50 эпох обучения происходит его уменьшение вдвое. Размер бача на обучающей выборке берется равным 8.

В процессе обучения модели оптимизируется функция потерь Dice-Loss:

$$DL(y, p) = 1 - \frac{2\sum_{i=1}^{N} y_i p_i}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} p_i + \varepsilon},$$

где $y_i (i = \overline{1, N})$−истинная метка класса пикселя i, $p_i (i = \overline{1, N})$− предсказання метка класса пикселя i, $\varepsilon = 10^{-6}$− число, добавленное в знаменатель второго слагаемого, чтобы гарантировать, что функция будет определена в случае, когда $y_i = p_i = 0 \ \forall i = \overline{1, N}$. Использование именно такой функции мотивировано тем, что знаменатель второго слагаемого функции учитывает общее количество граничных пикселей в глобальном масштабе, а числитель учитывает перекрытие между двумя наборами в локальном масштабе. Таким образом, функция Dice-Loss учитывает информацию о потерях как локально, так и глобально, что имеет решающее значение для высокой точности.

Качество решения моделями задачи сегментации людей будет оцениваться с помощью метрики Intersection over Union ($IoU$), которая показывает насколько маска человека, полученная с помощью нейросетей, совпадает с маской, выделенной вручную человеком.

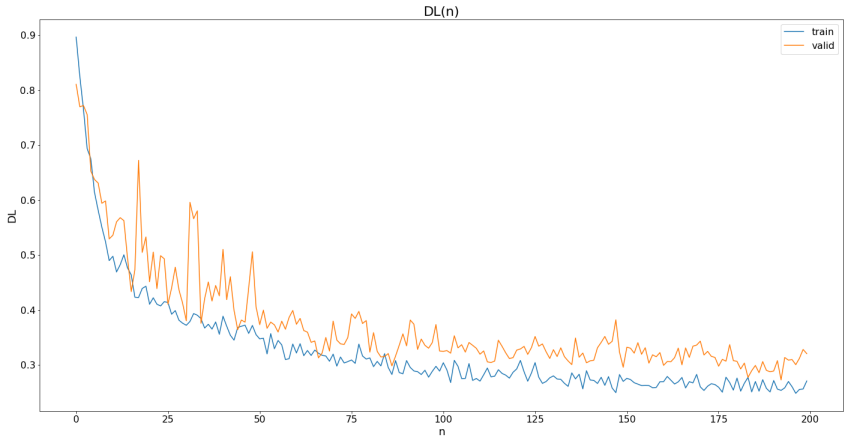


Рис. 6: График зависимости функции потерь $DL$ от эпохи с энкодером *ResNeXt-50* на обучающей и валидационной выборках

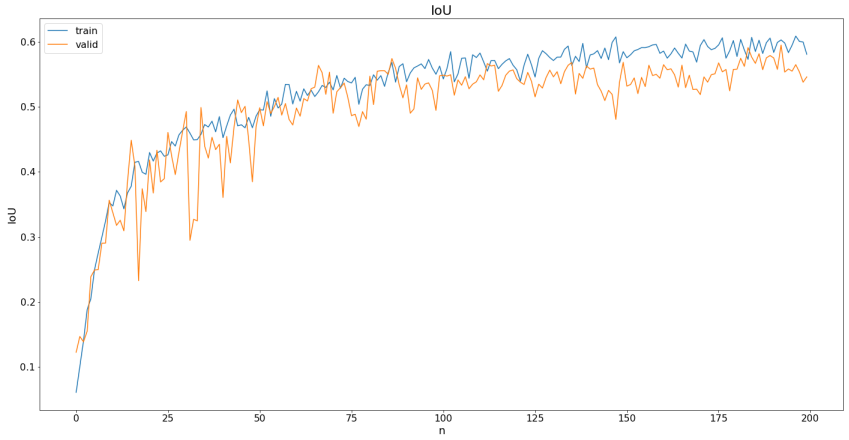


Рис. 7: График зависимости метрики $IoU$ от эпохи с энкодером *ResNeXt-50* на обучающей и валидационной выборках

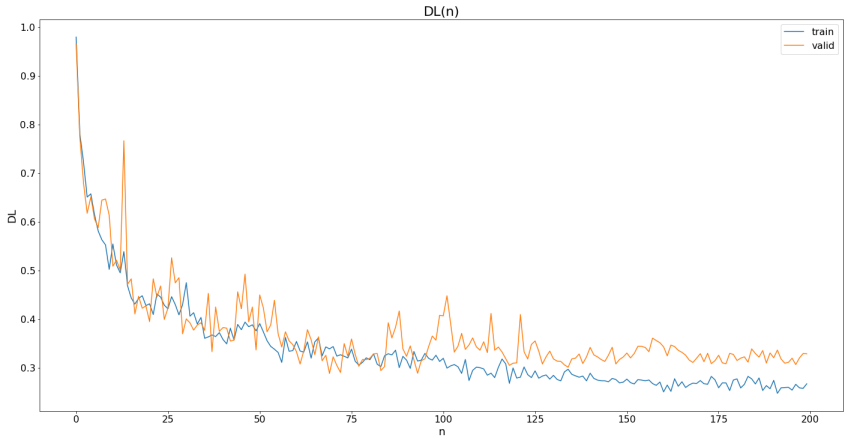


Рис. 8: График зависимости функции потерь $DL$ от эпохи с энкодером ResNet-152 на обучающей и валидационной выборках

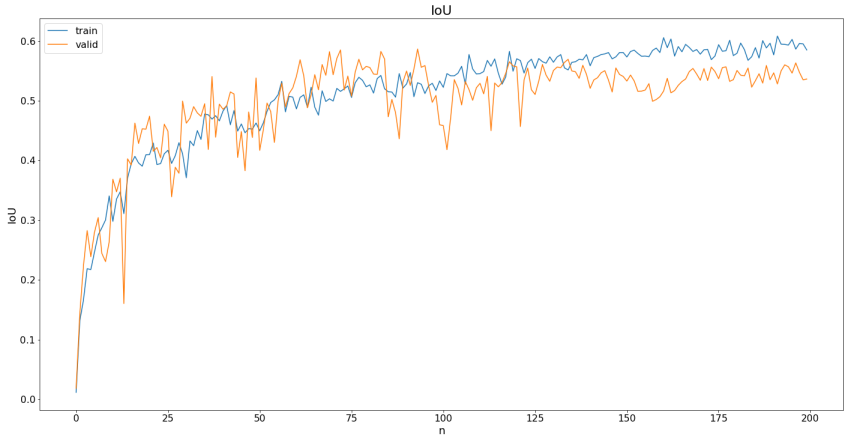


Рис. 9: График зависимости метрики $IoU$ от эпохи с энкодером ResNet-152 на обучающей и валидационной выборках

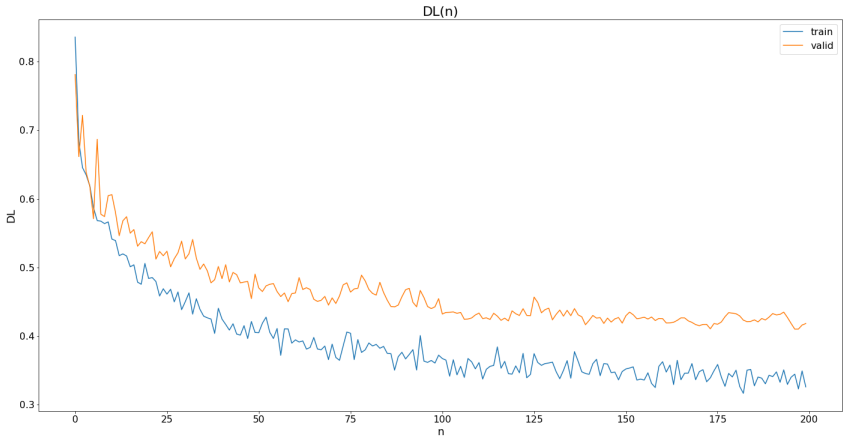


Рис. 10: График зависимости функции потерь $DL$ от эпохи с энкодером EfficientNet-B0 на обучающей и валидационной выборках

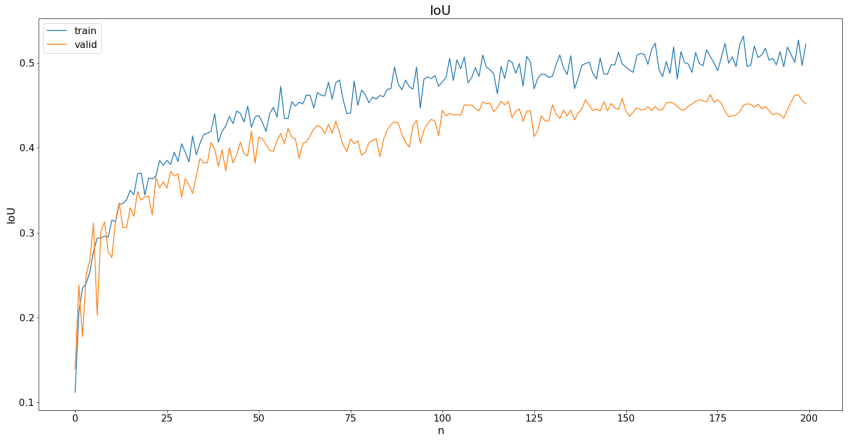


Рис. 11: График зависимости метрики $IoU$ от эпохи с энкодером EfficientNet-B0 на обучающей и валидационной выборках

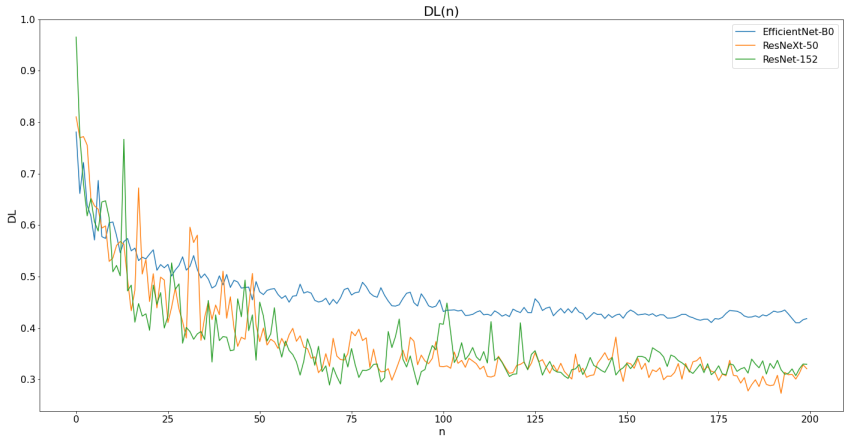


Рис. 12: График зависимости функции потерь $DL$ от эпохи на тестовой выборке

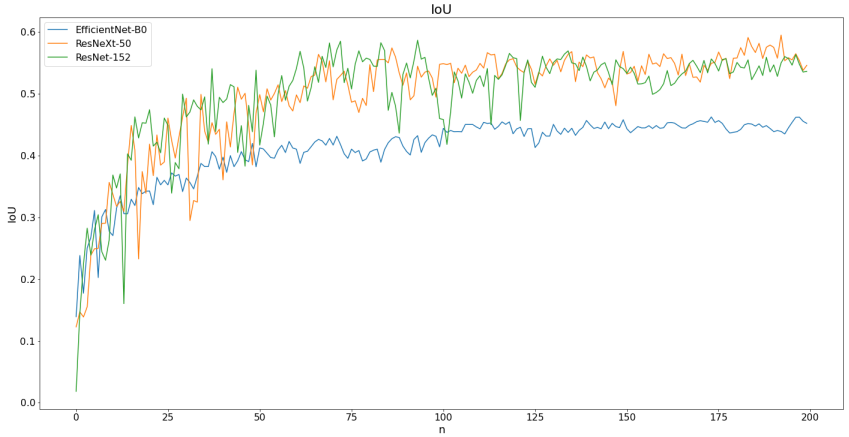


Рис. 13: График зависимости метрики $IoU$ от эпохи на тестовой выборке

В результате проведенных экспериментов оказалось, что в случае, когда используется энкодер EfficientNet-B0, модель демонстрирует результаты хуже чем в случае использования энкодеров *ResNeXt-50* и ResNet-152 (см. Рис. 12, Рис. 13). Это указывает на недостаточную для данной задачи сложность модели EfficientNet-B0. Наиболее качественную сегментацию осуществляет архитектура с энкодером *ResNeXt-50* , которая на тестовой выборке получает значение метрики $IoU = 0.43$ (см. Таблицу 4). Стоит отметить, что невысокие значения метрики $IoU$ обусловлены выбором в качестве декодера модели U-Net ([2]), которая ограничивает возможность нейросети точно сегментировать человеческие силуэты в ситуации, когда распределение людей на изображениях по всей выборке неоднородно.

| Энкодер | IoU |
|---|---|
| ResNeXt-50 | 0.4338 |
| ResNet-152 | 0.4115 |
| EfficientNet-B | 0.4008 |

Таблица 4: Сравнение качества сегментации на изображениях из CamVid

| Энкодер | Среднее время вывода (мс) |
|---|---|
| ResNeXt-50 | 17.305 |
| ResNet-152 | 30.812 |
| EfficientNet-B0 | 19.501 |

Таблица 5: Сравнение среднего времени вывода при используемых энкодерах

Примеры работы рассматриваемых архитектур на датасетах APSIS и CamVid приведены на Рис. 14 и Рис. 15. соотвественно. В эксперименте с датасетом APSIS все выбранные модели позволяют с высокой точностью ($IoU > 0.96$) сегментировать портреты людей.

Результаты замеров времени работы моделей представлены в Таблице 5. Наилучшую скорость работы показывает модель с энкодером *ResNeXt-50* , что очень важно для использования данной модели в мобильных устройствах. Эксперимент был проведен с использованием видеокарты NVIDIA GeForce RTX 2080 Ti.

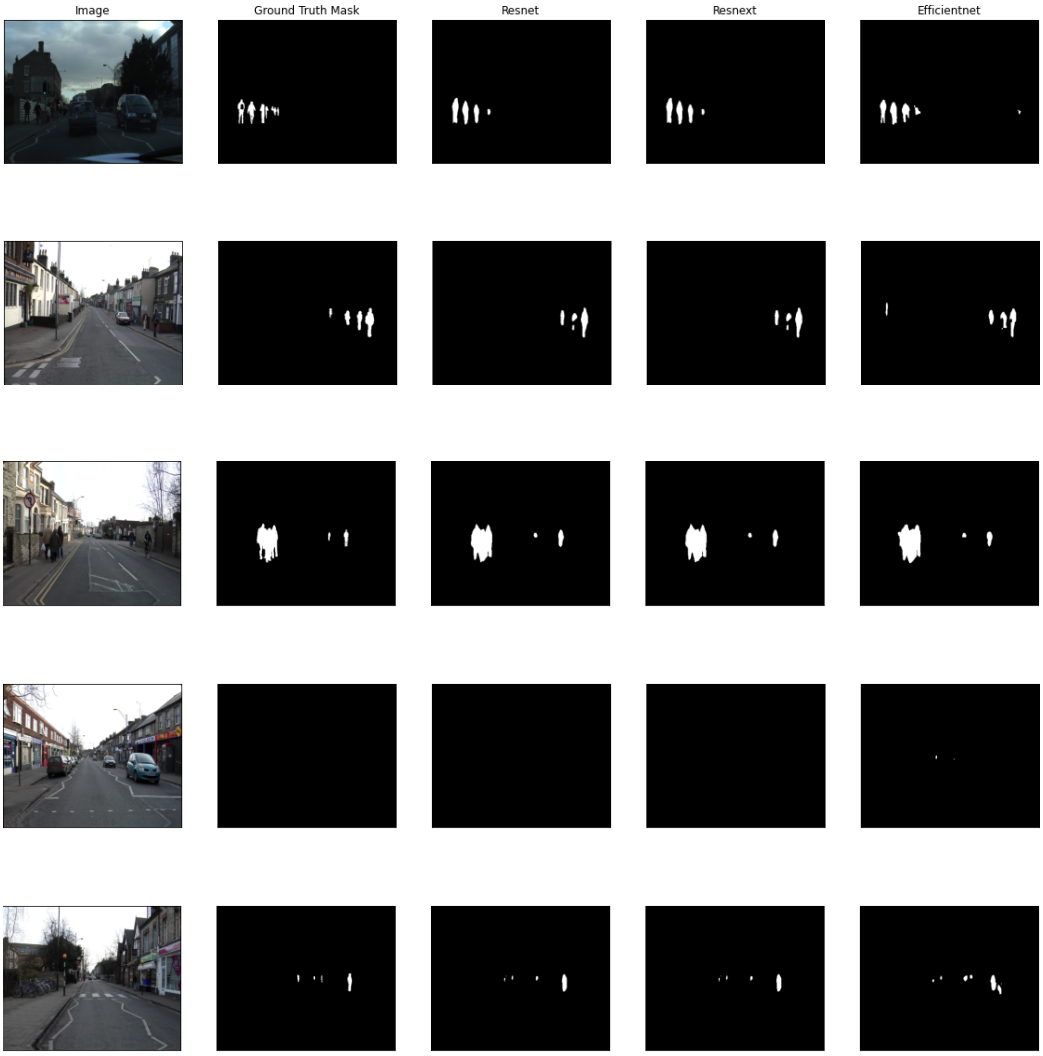


Рис. 14: Сравнение предсказанных масок людей с помощью различных архитектур на датасете CamVid

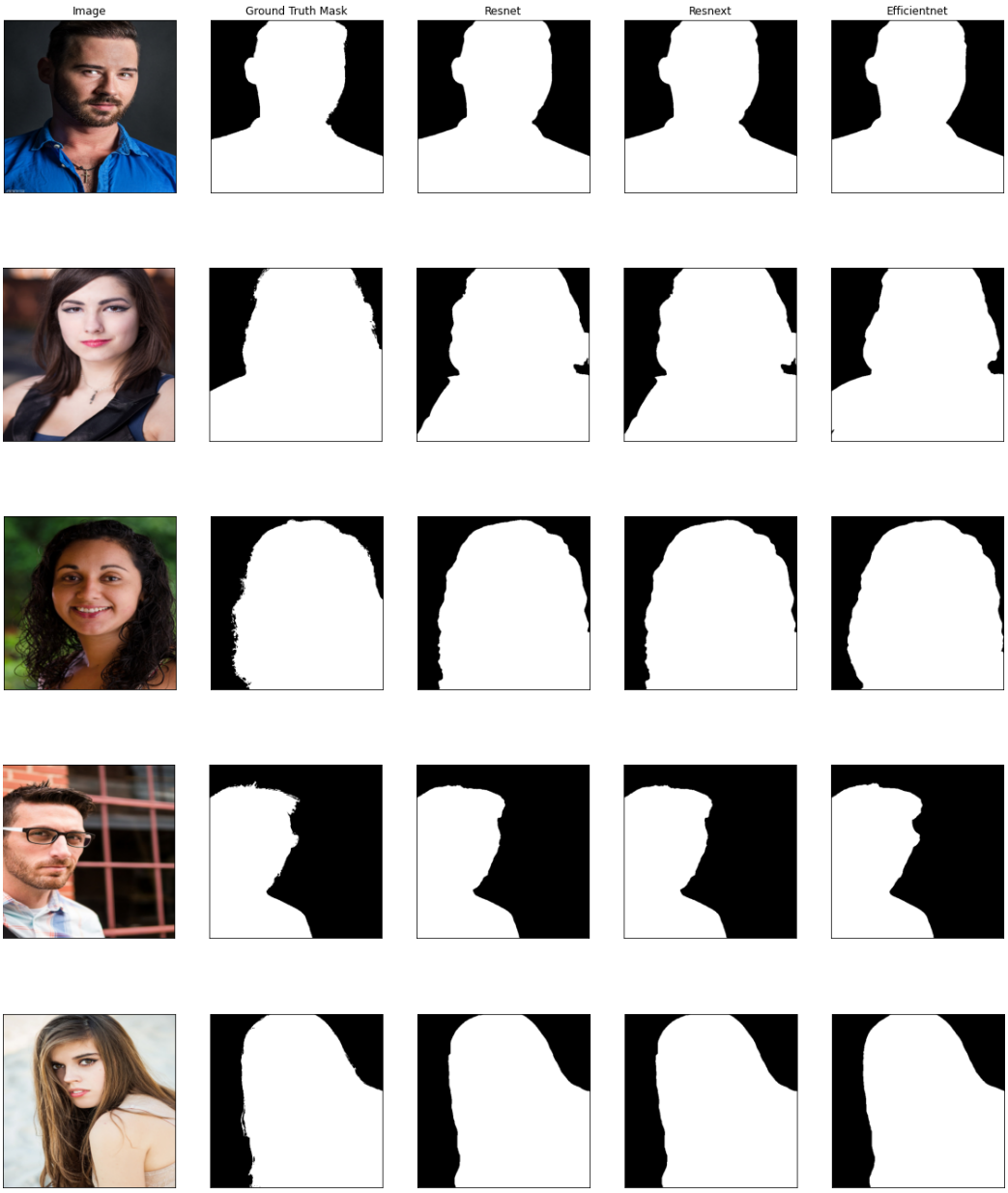


Рис. 15: Сравнение предсказанных масок людей с помощью различных архитектур на датасете APSIS

## 6. Заключение

В работе был проведен ряд экспериментов с различными энкодерами *ResNeXt-50* , ResNet-152 и EfficientNet-B0 для решения задачи сегментации человеческих фигур. В результате экспериментов было показано, что использование экстрактора признаков *ResNeXt-50* позволяет добиться компромисса между точностью и временем вывода модели, решающей задачу сегментации человеческих силуэтов. Более того, наилучшая скорость работы модели была также продемонстрирована с энкодером *ResNeXt-50* , имеющим сравнительно небольшое количество параметров и показывающим неизменно лучшее качество сегментации человеческих портретов, что делает данную модель наиболее предпочтительной для использования в мобильных устройствах по сравнению с энкодерами ResNet-152 и EfficientNet-B0. Идея использовать экстракторы признаков, предварительно обученные на большом наборе данных, таком как ImageNet, в эксперименте с датасетом APSIS показала свою состоятельность. Все рассматриваемые модели сумели достичь высокой точности ($IoU > 0.96$) сегментации портретов людей. В эксперименте с датасетом CamVid была установлена необходимость дополнительного подбора декодера в качестве альтернативы модели U-Net([2]). Данная проблема будет являться предметом будущих исследований.

## Список литературы


[1] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, 2017. arXiv:1611.05431.

[2] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, CoRR abs/1505.04597 (2015). URL: http://arxiv.org/abs/1505.04597. arXiv:1505.04597.

[3] M. Tan, Q. V. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, 2020. arXiv:1905.11946.

[4] H. Park, L. L. Sjösund, N. Monet, Y. Yoo, N. Kwak, Sinet: Extreme lightweight portrait segmentation networks with spatial squeeze modules and information blocking decoder, arXiv preprint arXiv:1911.09099 (2019).

[5] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, 2015. arXiv:1411.4038.

[6] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012) 1097–1105.

[7] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

[8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9.

[9] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, 2016. arXiv:1511.00561.

[10] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, 2015. arXiv:1505.04597.

[11] V. Iglovikov, A. Shvets, Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation, 2018. arXiv:1801.05746.

[12] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2015. arXiv:1409.1556.

[13] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, 2015. arXiv:1512.03385.

[14] R. Shah, S. Patil, A. Malhotra, R. Asati, A survey to study about different convolutional neural network on various image classifications, SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology 12 (2020) 236–242.

[15] S. Luo, H. Dai, L. Shao, Y. Ding, C4av: Learning cross-modal representations from transformers, in: European Conference on Computer Vision, Springer, 2020, pp. 33–38.

[16] Z. Wu, C. Shen, A. Van Den Hengel, Wider or deeper: Revisiting the resnet model for visual recognition, Pattern Recognition 90 (2019) 119–133.

[17] H. Touvron, A. Vedaldi, M. Douze, H. Jégou, Fixing the train-test resolution discrepancy: Fixefficientnet, arXiv preprint arXiv:2003.08237 (2020).

[18] Z. Müftüoğlu, M. A. Kizrak, T. Yildlnm, Differential privacy practice on diagnosis of covid-19 radiology imaging using efficientnet, in: 2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), IEEE, 2020, pp. 1–6.

[19] B. Yang, G. Bender, Q. V. Le, J. Ngiam, Condconv: Conditionally parameterized convolutions for efficient inference, arXiv preprint arXiv:1904.04971 (2019).

[20] X. Shen, A. Hertzmann, J. Jia, S. Paris, B. Price, E. Shechtman, I. Sachs, Automatic portrait segmentation for image stylization, in: Computer Graphics Forum, volume 35, Wiley Online Library, 2016, pp. 93–102.

[21] N. Ketkar, Introduction to pytorch, in: Deep learning with python, Springer, 2017, pp. 195–208.

[22] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017. `arXiv:1412.6980`.