

Нейросетевые решения на основе архитектуры U-Net для автоматического определения контуров натуральных кож

Е.А. Рыжкова, А.А.Казначеева, А.М. Казанцева

*Российский государственный университет имени А.Н. Косыгина
(Технологии. Дизайн. Искусство), г. Москва*

Аннотация: Целью данной работы является исследование применимости архитектуры U-Net для автоматического определения контуров натуральных кож с использованием библиотек TensorFlow и Keras на языке Python. Разработано программное приложение, основанное на методах, включающих библиотеки OpenCV, а также модель реализации глубокой сверточной нейронной сети. Набор данных для обучения и тестирования сети создавался с использованием аугментации. Обучение производилось методом стохастического градиентного спуска после разбиения выборки данных на тренировочные и тестовые изображения. В дальнейшем полученные результаты будут положены в работу над созданием автоматизированной системы, которая позволит определять контуры кожи и ее пороки, что, в свою очередь, откроет возможности к вычислению полезной площади кожи и созданию автоматизированной раскладки лекал с учетом выявленных пороков.

Ключевые слова: компьютерное зрение, определение контуров, натуральная кожа, машинное обучение, сверточные нейронные сети, U-Net архитектура, глубокое обучение.

За последние годы компьютерное зрение стало ключевой технологией во многих областях. В автоматизации промышленности компьютерное зрение повсеместно применяется для контроля качества и технологических процессов. При решении задач, в том числе, определения контуров и пороков натуральных кож, используют ряд методов распознавания объектов по изображениям, где большое значение имеет фильтрация. Фильтрация позволяет уменьшить влияние помех и посторонних объектов на изображении, а также характеризуется наличием шумов, возникающих при передаче изображения на обработку [1].

На сегодняшний день известно большое количество различных видов фильтров [2]. Один из таких фильтров, позволяющий определить контур натуральной кожи, был разработан на основе метода Кенни (Canny), который является одним из эффективных способов обработки изображений. Суть метода заключается в извлечении полезной структурной информации из n -го количества объектов, что значительно сокращает объем обрабатываемых

данных. Другой способ распознавания контура объекта основывается на применении стандартных функций `findContour()` и `drawContour()` библиотеки `OpenCV` на языке программирования `Python` [3].

Таким образом, можно сказать, что функции библиотек прекрасно подходят для решения задачи нахождения контура кожи. Тем не менее, существует ряд недостатков, в том числе определение и подбор максимального и минимального значений цветовых порогов вручную, а нахождение контуров кожи или дефектов из-за своего многообразия могут потребовать доработок алгоритма и новых надстроек. Для решения задачи повышения эффективности распознавания контуров необходимо применение методов машинного обучения и нейросетевых технологий [3].

Первоначальная цель разработки искусственных нейронных сетей (ИНС) состояла в том, чтобы имитировать процессы, происходящие в зрительной системе человека. Очевидно, что ИНС должна состоять из нескольких слоев, каждый из которых изменяет данные сначала локально, а затем все большими наборами нейронов, пока не будут выполнены задачи распознавания. Поэтому ИНС, как правило, предназначались для выполнения только классического процесса распознавания и получали входные данные от детекторов признаков, применяемых на предыдущих уровнях, не связанных с обучением [4].

Однако с течением времени возникла потребность в полномасштабном анализе реальных сцен, которые могли бы содержать изображения многих типов объектов во многих положениях [5]. Таким образом, нарастала потребность в переходе к гораздо более сложным многоуровневым системам распознавания, для которых ранние модификации ИНС были непригодны. Также появилась необходимость в обучении самой системы предварительной обработки, чтобы она точно соответствовала требованиям следующей системы анализа объектов; иными словами, возникла необходимость в

создании интегрированных многослойных нейронных сетей [6].

Новым типом архитектуры стала глубокая сверточная нейронная сеть (convolutional neural network, CNN). Во многих отношениях это была менее требовательная архитектура, поскольку: 1) каждый нейрон CNN не должен быть подключен ко всем выходам предыдущего слоя нейронов; 2) нейроны имеют одинаковые весовые параметры по всему слою. Еще одна особенность CNN заключается в том, что она использует нелинейную функцию активации ReLU, которая означает «Rectified Linear Unit» (спрямленный линейный блок) и определяется как:

$$f(x) = \max(0, x), \quad (1)$$

где x – выходное значение непосредственно предшествующего слоя свертки. Эта функция ценна тем, что требует меньше вычислений, чем сигмовидная функция, и меньше искажает большие сигналы.

На сегодняшний день CNN применяют в таких задачах, как распознавание речи, обработки аудиосигналов, обработки временных рядов, для анализа смысла текстов и т.д. CNN получают на вход один вектор и преобразуют его через ряд скрытых слоев.

В настоящее время известен целый ряд архитектур глубоких сверточных нейронных сетей. Одной из самых популярных является модель VGG, разработанная в Оксфордском университете в Группе визуальной геометрии (Visual Geometry Group). Модель содержит две конфигурации на 16 и 19 слоев, где впервые была применена идея рассматривать фильтры размером 3×3 (вместо 7×7 и 11×11 , использовавшихся ранее) с единичным шагом свертки. Такая концепция позволяет увеличить количество слоев, при этом одновременно уменьшая общее число весов. Еще одним преимуществом данной модели является наличие дополнительной нелинейности между слоями, что позволяет увеличить «разрешающую способность» по сравнению с единственным слоем с большей сверткой.

Еще одной важной архитектурой является Inception. Она создана в компании Google в 2014 году. Имеет глубину в 22 слоя и структуру, состоящую из модулей, комбинирующих свертки размером 1×1 , 3×3 и 5×5 . Каждый модуль представляет собой набор из четырех «маленьких» сетей, выходы которых объединяются в выходные каналы и передаются на следующий слой. Одно из ключевых нововведений – использование сверточных слоев 1×1 не столько в качестве дополнительной нелинейности, сколько для понижения размерности между слоями. Свертки 3×3 и тем более 5×5 между слоями с большим числом каналов, оказываются крайне ресурсоемкими, несмотря на малые размеры отдельно взятых фильтров. А фильтры 1×1 могут помочь сократить число каналов, прежде чем подавать их на фильтры большего размера [6].

Модель ResNet представлена командой из Microsoft Research в 2015 году для идеи глубокого остаточного обучения (deep residual learning). В базовой структуре слои, идут последовательно друг за другом. Отдельные уровни, составные блоки сети выглядят как сверточные слои, обычно с дополнительной нормализацией. Отличие от предыдущих архитектур заключается в том, что в остаточном блоке слой из нейронов можно «обойти»: есть специальная связь между выходом предыдущего слоя $x^{(k)}$ и следующего слоя $x^{(k+1)}$.

Математически это выглядит следующим образом:

$$y^{(k)} = F(x^{(k)}) + x^{(k)}, \quad (2)$$

где $x^{(k)}$ – входной вектор слоя k , $F(x)$ – функция, которую вычисляет слой нейронов, а $y^{(k)}$ – выход остаточного блока, который потом станет входом следующего слоя $x^{(k+1)}$.

Таким образом, если блок в целом должен аппроксимировать функцию $H(x)$, то это достигается тогда, когда $F(x)$ аппроксимирует остаток (residue)

$H(x)-x$, отсюда и название «остаточные сети» (residual networks). В остаточном блоке слой нейронов обучается воспроизводить изменения входных значений, необходимые для получения итоговой функции. Считается, что обучить «остаточную» функцию проще, чем исходную [7].

Одной из стандартных архитектур CNN является U-Net, разработанная учеными Фрайбургского университета Германии в 2015 году для решения задач сегментации биомедицинских изображений. Имеет симметричную U-образную структуру, благодаря которой получила свое название. В отличие от традиционных нейросетевых моделей, которые используют в составе полносвязные слои, U-Net задействует экстраполирование недостающих данных в приграничных областях изображения, зеркально отражая входное изображение.

Сеть состоит из четырех градаций. Каждая градация имеет связь с пропуском идентичности между операциями шаговой свертки 2×2 понижения разрешения и транспонированной свертки 2×2 повышения разрешения. Количество каналов в каждом слое от первой до четвертой градации составляет 64, 128, 256 и 512 соответственно. В каждой градации уменьшения и увеличения разрешения задействовано четыре последовательных обходных блока. Каждый обходной блок содержит только одну функцию активации ReLU. Всего сеть содержит 23 сверточных слоя. Процесс обучения осуществляется сквозным способом на небольшом количестве изображений. Кроме того, сеть демонстрирует высокую скорость работы: сегментация изображения 512×512 пикселей занимает менее секунды на современном графическом процессоре [8].

Данный метод позволяет не только определить класс изображения целиком, но и сегментировать его области по классу, т.е. создать маску, которая будет разделять изображение на несколько классов.

Оператор свертки вычисляется по формуле:

$$(L * S)_{xy} = \sum_{i=1}^k \sum_{j=1}^n S_{ij} \times L_{x+i-1, y+j-1}, \quad (3)$$

где L – изображение размером $x \times y$, S – ядро свертки (матрица размерности $k \times n$).

Анализируя вышеперечисленные архитектуры глубоких сверточных нейросетей применительно к решению задачи определения контуров натуральных кож приходим к выводу, что наиболее эффективным методом является модель U-Net в виду своей простоты, универсальности и быстродействия.

В данной работе приведена реализация программного приложения LeatherEye, выполненное на базе нейронной сети с архитектурой U-Net на языке Python с применением библиотек TensorFlow и Keras. Программа функционирует в 6 режимах: 4 основных (задействованы в операциях по обнаружению контуров, в том числе с применением библиотеки OpenCV) и 2 вспомогательных. На рис. 1 представлено окно справочной системы и выбора режима работы программного приложения.

Добро пожаловать в Leather Eye v.02.01

```
----- СПРАВКА -----  
Данная программа может обнаружить контур кожи на изображении.  
Далее будет предложено ввести код режима работы, ниже приведены расшифровки режимов:  
Нахождение контура кожи OpenCV = 1  
Подготовка изображений для нейросети = 2  
(До)обучение нейронной сети = 3  
Проверка работы нейронной сети = 4  
Выход из программы = N  
Чтобы вызвать справку нажмите клавишу H  
-----
```

Введите код режима работы программы:

Рис. 1. – Справка и выбор режима работы программы

Нейросети требуют определенного количества и качества данных [9, 10]. Для задачи распознавания контура в рассматриваемом случае набор

данных создавался с использованием метода аугментации – процесса искусственного генерирования новых данных на основе существующих.

Для реализации аугментации был написан программный код, выполняющий процесс автоматически при помощи специально созданной библиотеки `imgaug`, из которой импортируется функция `augmenters`. Внутри цикла в пределах заданного количества углов, производился поворот изображения на соответствующий угол счетчика. Так же была сформирована маска для изображений, с соответствующими названиями. Исходные изображения и маски поворачивались на 1 градус столько раз, сколько потребуется изображений. На рис. 2 представлен результат выполнения аугментации.

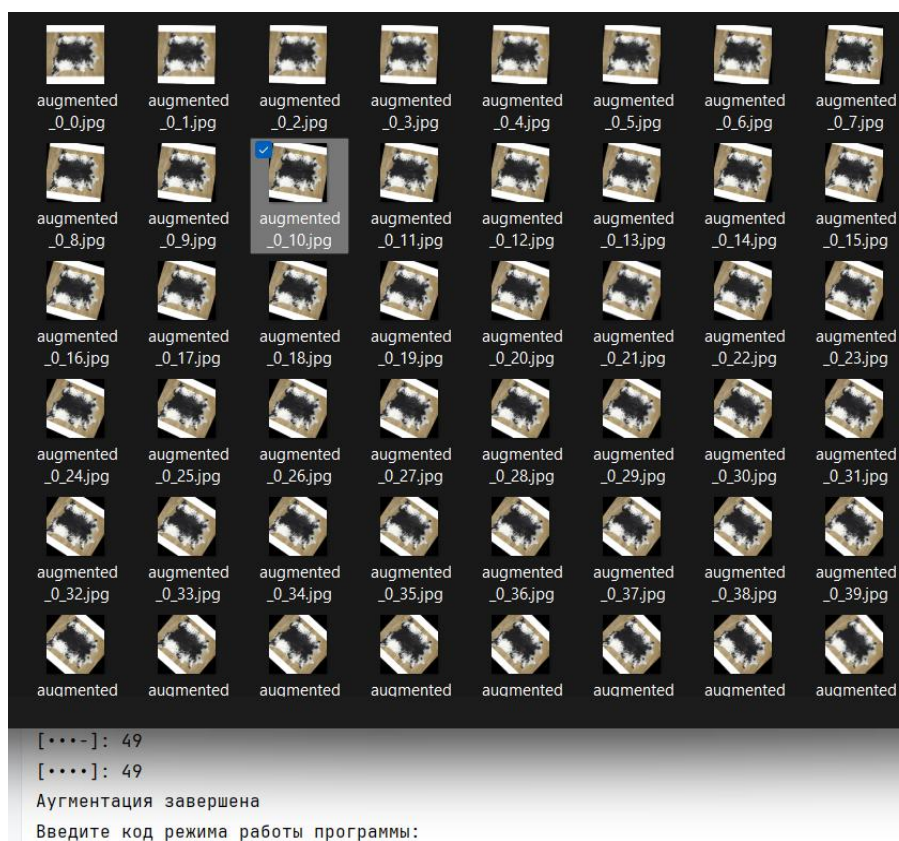


Рис. 2. – Результат выполнения аугментации изображения

Перед тем как обучить нейросеть необходимо проверить соответствие масок и изображений нужным требованиям. В нашем случае проверялись качество маски и соотношение маски изображению. Для проверки были

использованы библиотеки PIL и Matplotlib. В результате выполнения кода программы получены изображения, продемонстрированные на рис. 3.

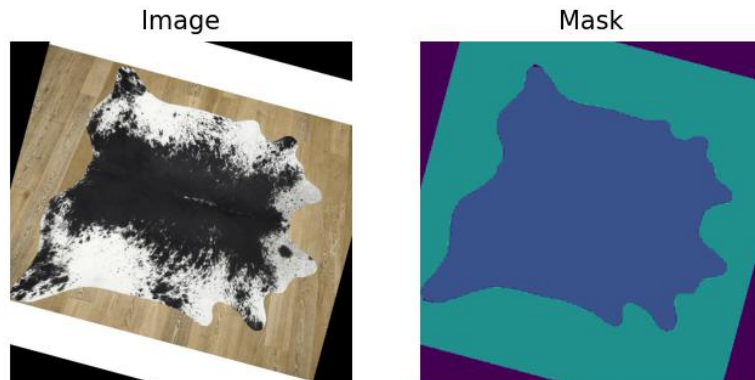


Рис. 3. – Пример вывода изображения и маски для проверки

Перед началом обучения программа производит загрузку изображений при помощи функции `load_data`. Пути к каталогам с изображениями образуются благодаря функции `os.listdir`, которая проходит по всем элементам каталога и возвращает имя файла, а далее объединяет имя файла с относительным и абсолютным путем к изображению и формирует массив путей.

Обучение нейросети производилось на основе входных изображений и соответствующих им масок методом стохастического градиентного спуска после случайного разбиения выборки данных на тренировочные (80%) и тестовые (20%) изображения размером 256×256 пикселей. Результат обучения представлен на рис. 4.





```
Введите имя модели (без расширения): model_demo
Вы хотите обучить модель? (Y\N): y
Введите количество эпох: 3
Epoch 1/3
1/1  19s 19s/step - accuracy: 0.9641 - loss: 0.0174 - precision: 0.0000e+00 - recall: 0.0000e+00
Epoch 2/3
1/1  14s 14s/step - accuracy: 0.9384 - loss: 0.0175 - precision: 0.0000e+00 - recall: 0.0000e+00
Epoch 3/3
1/1  13s 13s/step - accuracy: 0.9691 - loss: 0.0175 - precision: 0.0000e+00 - recall: 0.0000e+00
1/1  1s 1s/step - accuracy: 0.9275 - loss: 0.0177 - precision: 0.0000e+00 - recall: 0.0000e+00
```

Рис. 4. – Результат обучения нейросети

После обучения необходимо проверить корректность работы

нейронной сети на новом изображении и оценить результаты. После выбора соответствующего режима будет предложено выбрать имя файла модели без расширения для загрузки. Программа выведет структуру нейросети и параметры, в виде их количества, общего веса и других. Затем потребуется ввести относительный путь к изображению, с которым предстоит работать, а после имя файла без расширения (предполагается, что у изображения формат jpg). Как только ввод будет реализован, программа проведет сегментацию и выведет результат на экран, как показано на рис. 5.

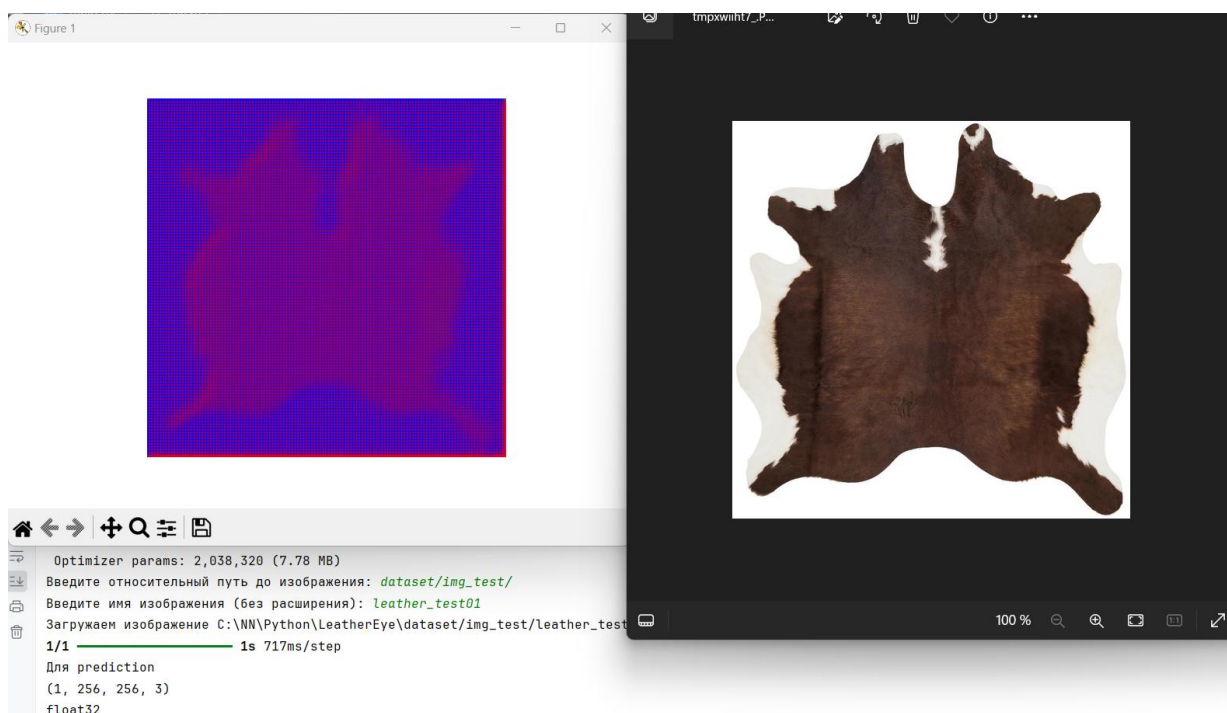


Рис. 5. – Результат работы нейросети

Таким образом разработано программное приложение LeatherEye по определению контуров натуральных кож на базе методов, включающих применение библиотеки OpenCV и глубокой сверточной нейронной сети. Проведено исследование применимости нейросети архитектуры U-Net для автоматического определения контуров натуральных кож. Установлено, что для решения поставленной задачи данная модель является наиболее эффективной в виду своей простоты, универсальности и быстродействия. В дальнейшем полученные результаты будут положены в работу над созданием

автоматизированной системы, которая позволит определять контуры кожи и ее пороки, что в свою очередь откроет возможность к вычислению полезной площади кожи и созданию автоматизированной раскладки лекал с учетом выявленных пороков.

Литература

1. Р. Гонсалес, Р. Вудс. Цифровая обработка изображений. // М: Техносфера, 2005. 1007 с.
 2. Nafstad O., Gronstol H. Variation in the Level of Grain Defect Light Flecks and Sports on Cattle Hides. // Acta vet. scand. 2001. pp. 91-98.
 3. Калачев Н.М., Казначеева А.А., Рыжкова Е.А. Применение нейросетевых технологий в задачах оценки качества натуральных кож. // Инновационное развитие техники и технологий в промышленности. Сборник материалов Всероссийской научной конференции молодых исследователей с международным участием. Москва. 2023. С. 187-191.
 4. Николенко С., Кадурич А., Архангельская Е. Глубокое обучение. // СПб.: Питер, 2018. 480 с.
 5. Крейман Г. Биологическое и компьютерное зрение. // ДМК-Пресс: М, 2022. 314 с.
 6. Дэвис Р., Терк М. Компьютерное зрение. Современные методы и перспективы развития. // М.: ДМК Пресс, 2022. 690 с.
 7. Heetal K. Deep Residual Learning for Image Recognition // Proc. 2016 CVPR. 2016. P. 770–778.
 8. Абдулина К.М., Спивак А.И. Применение нейронной сети архитектуры U-net для задачи локализации нарушений целостности цифровых изображений. // Научно-технический вестник информационных технологий, механики и оптики. 2020. № 3. URL: cyberleninka.ru/article/n/primenenie-neyronnoy-seti-arhitektury-u-net-dlya-zadachi-lokalizatsii-narusheniy-tselostnosti-tsifrovyyh-izobrazheniy.
-

9. Казначеева А.А., Власенко О.М., Захаркина С.В., Ступак М.М. Применение нейросетевых технологий в задачах контроля качества изделий текстильной промышленности. // Инженерный вестник Дона. 2024. №1. URL: ivdon.ru/ru/magazine/archive/n1y2024/8947.

10. Murashko F.V., Ryzhkova E.A., Vlasenko O.M. Search for an object in an image by image difference method to find contours of a natural leather blank in pattern cutting processs. *Fibre Chemistry*. 2018. Т. 50. № 1. pp. 38-41.

References

1. R. Gonsales, R. Vuds. *Cifrovaya obrabotka izobrazhenij* [Digital image processing]. M: Tekhnosfera, 2005. 1007 p.

2. Nafstad O., Gronstol H. Variation in the Level of Grain Defect Light Flecks and Sports on Cattle Hides. *Acta vet. scand.* 2001. pp. 91-98.

3. Kalachev N.M., Kaznacheeva A.A., Ryzhkova E.A. *Sbornik materialov Vserossijskoj nauchnoj konferencii molodyh issledovatelej s mezhdunarodnym uchastiem*. Moskva. 2023. pp. 187-191.

4. Nikolenko S., Kadurin A., Arhangel'skaya E. *Glubokoe obuchenie* [Deep learning]. SPb.: Piter, 2018. 480 p.

5. Krejman G. *Biologicheskoe i komp'yuternoe zrenie* [Biological and computer vision]. DMK-Press: M, 2022. 314 p.

6. Devis R., Terk M. *Komp'yuternoe zrenie. Sovremennye metody i perspektivy razvitiya* [Advanced Methods and Deep Learning in Computer Vision]. M.: DMK-Press, 2022. 690 p.

7. Heetal K. Deep Residual Learning for Image Recognition. *Proc. 2016 CVPR*. 2016. pp. 770–778.

8. Abdulina K.M., Spivak A.I. *Nauchno-tehnicheskij vestnik informacionnyh tekhnologij, mekhaniki i optiki*. № 2020. 3. URL: cyberleninka.ru/article/n/primeneniye-neyronnoy-seti-arhitektury-u-net-dlya-zadachi-lokalizatsii-narusheniy-tselostnosti-tsifrovyyh-izobrazheniy.



9. Kaznacheeva A.A., Vlasenko O.M., Zaharkina S.V., Stupak M.M. Inzhenernyj vestnik Dona. 2024. №1. URL: ivdon.ru/ru/magazine/archive/n1y2024/8947.

10. Murashko F.V., Ryzhkova E.A., Vlasenko O.M. Fibre Chemistry. 2018. T. 50. № 1. pp. 38-41.

Дата поступления: 27.10.2024

Дата публикации: 5.12.2024