

## Методика подготовки набора данных для обучения и тестирования нейронных сетей в среде MATLAB

*А.В. Ковалев<sup>1</sup>, А.С. Исаева<sup>2</sup>*

*<sup>1</sup>Научно-конструкторское бюро моделирующих и управляющих систем (НКБ МИУС) ЮФУ*

*<sup>2</sup>Дизайн-центр микроэлектронной компонентной базы для систем искусственного интеллекта ЮФУ*

**Аннотация:** В работе предлагается методика подготовки набора данных для обучения и тестирования нейронных сетей в среде MATLAB с применением пакета функций Image Processing Toolbox. Приводится пример реализации данной методики для подготовки набора данных для обучения сверточных нейронных сетей классифицировать семена пшеницы на качественные и некачественные по их изображениям. Данная методика была использована для получения набора данных, состоящих из 900 изображений, успешно использованных для обучения и тестирования сверточной нейронной сети.

**Ключевые слова:** набор данных, сверточные нейронные сети, искусственный интеллект, классификация изображений.

Алгоритмы распознавания изображений, использующие нейронные сети, являются в настоящее время самым эффективным средством автоматизации анализа и классификации изображений. Большую популярность получили методы классификации на основе сверточных нейронных сетей - алгоритма глубокого обучения, который может принимать входное изображение, присваивать важность (веса и смещения) различным аспектам изображения и отличать одно от другого [1-3]. При этом изображения в сравнении с другими алгоритмами требуют гораздо меньшей предварительной обработки. Эффективность обучения напрямую зависит от доступа к нужному количеству аннотированных однотипных изображений определенных объектов. Несмотря на наличие и активное развитие проектов по созданию и сопровождению массивных баз данных аннотированных изображений (ImageNet, VisualData, Labelme и другие [4]), далеко не всегда получается подобрать оптимальный набор данных для конкретной разработки и в этом случае приходится формировать набор данных

---

самостоятельно. Методикам формирования набора данных посвящено большое количество научных работ, анализируя которые, можно выделить следующие подходы: сегментация размещенных в открытом доступе фотографий, содержащих нужные объекты [5], фотографирование или видеосъемка процессов с последующей сегментацией [6], использование программных алгоритмов для искусственного расширения имеющейся базы данных [7], использование программных алгоритмов для генерации трехмерных объектов и получение большого количества изображений, например, с различных ракурсов и уровней освещенности [8], использование программ для синтетической генерации составных изображений из набора имеющихся [9]. В данной работе предлагается методика подготовки набора данных для обучения и тестирования нейронных сетей, основанная на обработке полученных самостоятельно фотографий в среде MATLAB с применением пакета функций Image Processing Toolbox.

Классификация однотипных объектов (в том числе, пищевых продуктов, см. например, [10]) на классы в зависимости от их качества, наряду с распознаванием лиц, одна из самых востребованных задач для нейросетей в настоящее время. В данной работе методика приводится применительно к подготовке набора данных для обучения сверточных нейронных сетей классифицировать семена пшеницы на качественные и некачественные по их изображениям. Для обработки было выбрано изображение, приведенное на рис. 1.

Изображение представляет собой матрицу размерами  $3000 \times 4000 \times 3$ , загружается с помощью функции `imread`. Каждая ячейка матрицы — это отдельный пиксель изображения. Первые два измерения соответствуют размерам изображения в пикселях, а третье измерение показывает, что присутствуют три канала яркости: красный, зеленый и синий каналы.



Рис.1. – Изображение, обрабатываемое для разработки методики подготовки набора данных для обучения и тестирования нейронных сетей в среде MATLAB с применением пакета функций Image Processing Toolbox

Обращаясь к нужному слою матрицы, можно разбить входное RGB изображение на отдельные каналы. В результате будут получены три матрицы размерами  $3000 \times 4000$ . Визуализация полученных матриц представлена на рис.2.

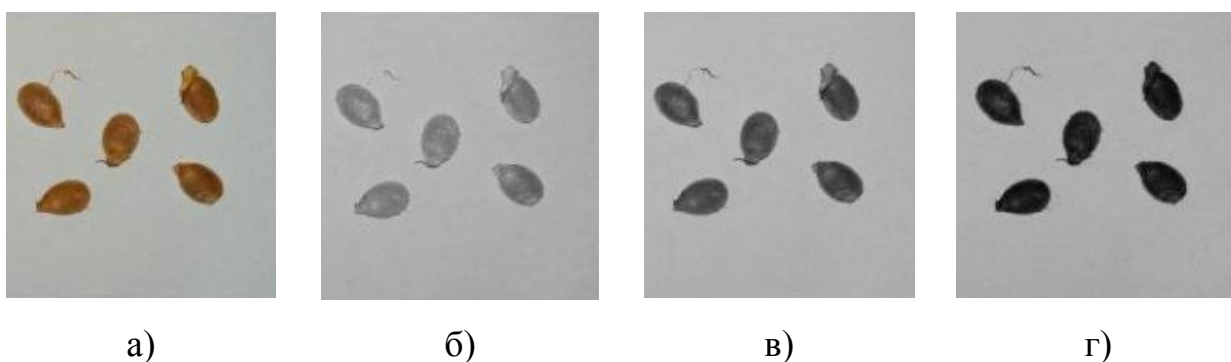


Рис. 2. – Входное RGB изображение (а) и изображение каждого канала: красный (б), зеленый (в) и синий (г)

Полученные с помощью разбивки по каналам изображения были подвергнуты пороговой бинаризации с помощью функции `im2bw`. Суммарное бинарное изображение было получено объединением бинарных изображений для каждого канала яркости, они представлены на рис. 3. Поскольку яркость изображения в каждом канале различна, то и объекты на бинарных изображениях выделены с разной успешностью. Однако на суммарном изображении все объекты четко просматриваются. Многоступенчатый подход получения бинарного изображения был реализован потому, что прямая бинаризация полноцветного входного изображения при одинаковых параметрах приводит к меньшей четкости изображения, что можно наблюдать на рис. 3, д.

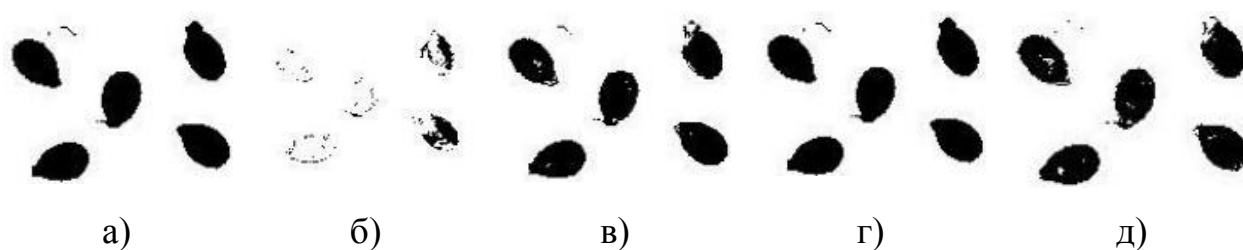


Рис.3. – Бинарные изображения: суммарное бинарное изображение (а), бинарное изображение красного канала (б), бинарное изображение зеленого канала (в), бинарное изображение синего канала (г), бинарное изображение, полученное из итогового (д)

Для проведения морфологической обработки изображения и обнаружения объектов на нем, необходимо получить белые изображения объектов на черном фоне, для чего было вычислено комплементарное дополнение к бинарному изображению с помощью функции `imcomplement`. Пустоты на изображении – области с нулевым значением, полностью окруженные пикселями со значением «единица» были заполнены с использованием функции `imfill`. Полученное изображение содержит шум, а

границы объектов неровные. Для удаления шума и сглаживания границы объектов применялись морфологические операции, а именно - функция морфологического открытия изображения `imopen` с определенным структурным элементом. Структурный элемент — это форма, которая используется для обработки. Форма элемента может быть выбрана из предопределенных или задана самостоятельно. Размер элемента задается, исходя из размеров объектов на изображении в пикселях. Объекты размером меньше удвоенного размера структурного элемента удаляются из дальнейшего анализа. На рис. 4 представлено комплементарное дополнение к бинарному изображению и результат морфологической обработки.

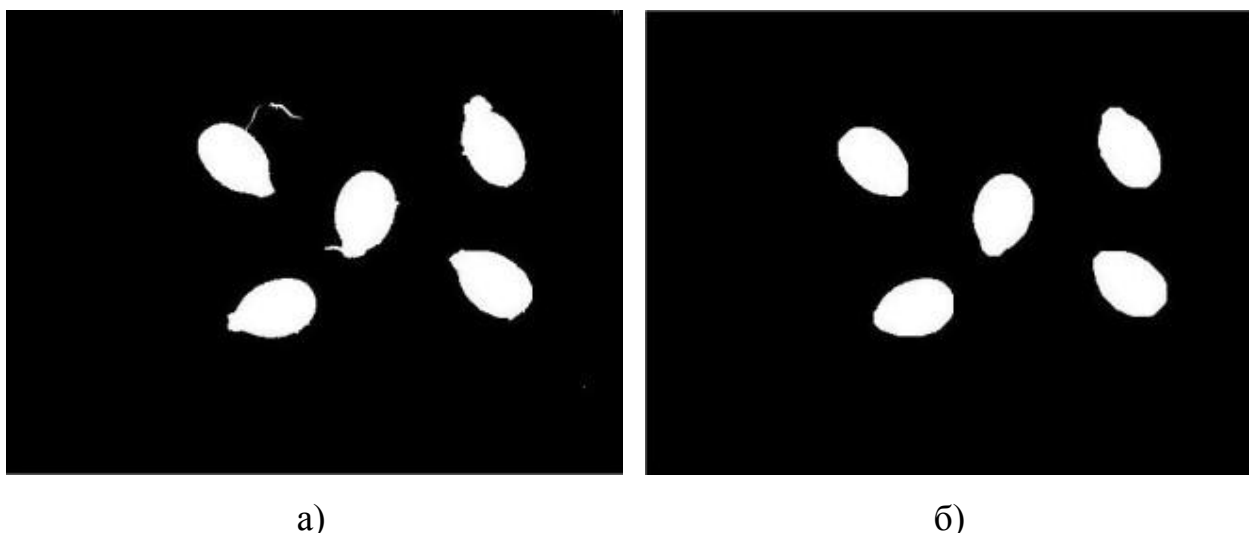


Рис.4. – Комплементарное дополнение к бинарному изображению (а) и результат морфологической обработки (б)

Таким образом было получено корректное бинарное изображение, пригодное для дальнейшего анализа. Для работы с признаками объектов их анализа и обработки используется функция `regionprops`. Она позволяет извлечь информацию о бинарном изображении, такую, как площадь объектов, их центры и тому подобное. В результате применения приведенной функции на исходном изображении были выделены и пересчитаны

---

найденные объекты. Фрагмент программного кода и результат обнаружения объектов на исходном изображении представлены на рис. 5.

```
%%  
imgregion = regionprops (img_opened, 'centroid');  
[labeled, numObjects] = bwlabel (img_opened,4);  
stats = regionprops (labeled, 'Eccentricity', 'Area', 'BoundingBox');  
eccentricities = [stats.Eccentricity];  
%%  
idxofskittles = find(eccentricities);  
statsDefects = stats (idxofskittles);  
figure, imshow (img);  
hold on;  
for idx = 1:length (idxofskittles)  
    h = rectangle ('Position', statsDefects (idx).BoundingBox);  
    set (h, 'EdgeColor', [1 0 0]);  
    hold on;  
end  
title (['Найдено ', num2str (numObjects), ' объектов']);
```



а)

б)

Рис.5. – Фрагмент программного кода (а) и результат обнаружения объектов на исходном изображении (б)

Для работы и обучения нейронной сети необходимо вырезать найденные объекты из общего изображения и сохранить их. Поскольку информация о количестве объектов и о центре каждого найденного объекта уже известна, было реализовано автоматическое вырезание объекта из общего изображения и сохранение вырезанного фрагмента в качестве отдельного файла. Для вырезания найденных объектов используется функция `imcrop`. Запись и сохранение фрагмента в отдельный файл осуществляется с помощью функции `imwrite`. Фрагменты могут сохраняться как отдельными файлами, так и одним многостраничным `tif` файлом, который будет содержать все вырезанные фрагменты. Данная операция была автоматизирована с помощью цикла. Результатом выполнения описанной методики является `tif` файл, содержащий все найденные на исходном изображении и вырезанные из него объекты и ряд пронумерованных `jpeg` файлов, содержащих вырезанный объект. Результирующие файлы представлены на рис. 6.

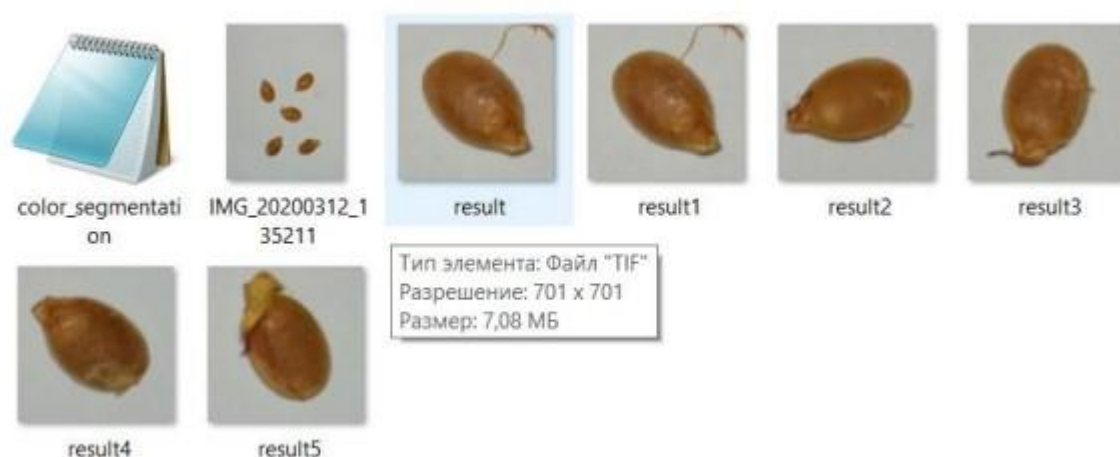


Рис.6. – Файлы, полученные в результате реализации методики подготовки набора данных для обучения и тестирования нейронных сетей

На основе описанной выше методики, из фотографий семян пшеницы был создан набор данных для обучения НС, содержащий 900 изображений семян пшеницы. Фотографии были сделаны на фотоаппарат, установленный на штативе, фокусное расстояние объектива - 16 см., расстояние между краем насадки объектива и плоскостью с семенами - 30 см. Для фотографий, на которые, помимо изображений семян, попали изображения посторонних объектов, была реализована функция вырезания рабочей области `imcrop`. Для неподвижно закрепленной камеры процесс выделения рабочей области можно автоматизировать для всех обрабатываемых изображений. В результате выполнения алгоритма были выделены все семена, находящиеся в выбранной рабочей области. Однако, из-за близкого расположения отдельных семян друг к другу, наличия затенений, некоторые группы семян выделялись как один объект. Такие случаи были сохранены отдельно для дальнейшего анализа. Повышение порога бинаризации уменьшает количество групповых объектов, но хорошо освещенные семена выделяются не полностью или вовсе не обнаруживаются. Каждый найденный объект был вырезан окном размером  $90 \times 90$  пикселей и сохранен в качестве отдельного файла. Был написан отдельный скрипт MATLAB, позволяющий производить

масштабирование созданных наборов изображений. Для масштабирования требуется указать количество изменяемых изображений, путь к ним вместе с общей частью имени, путь к папке, в которую будет произведено сохранение изображений с новыми размерами и непосредственно размеры изображения в пикселях. Набор данных, полученных по описанной в работе методике, был успешно применен при разработке способа анализа и оценки качества семян пшеницы посредством классификации их изображений с использованием искусственного интеллекта, а именно - сверточной нейронной сети.

*Научные исследования выполнены в рамках проекта «Создание высокотехнологичного производства аппаратно-программных комплексов обработки сельхозсырья на основе СВЧ излучения» (Соглашение с Минобрнауки РФ № 075-11-2019-083 от 20.12.2019 г., Договор ЮФУ № 18 от 20.09.2019 г., номер работы в ЮФУ № ХД/19-25-РТ).*

### Литература

1. Соловьев Р.А., Тельпухов Д.В., Кустов А.Г. Автоматическая сегментация спутниковых снимков на базе модифицированной сверточной нейронной сети UNET // Инженерный вестник Дона, 2017, №4. URL: [ivdon.ru/ru/magazine/archive/n4y2017/4433](http://ivdon.ru/ru/magazine/archive/n4y2017/4433).

2. Галушка В.В., Фатхи В.А. Формирование обучающей выборки при использовании искусственных нейронных сетей в задачах поиска ошибок баз данных // Инженерный вестник Дона, 2013, №2. URL: [ivdon.ru/uploads/article/pdf/IVD\\_7\\_Galushka.pdf\\_1597.pdf](http://ivdon.ru/uploads/article/pdf/IVD_7_Galushka.pdf_1597.pdf).

3. Li C., Zhang Y., Qu, Y. Object detection based on deep learning of small samples // 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI), 2018, pp. 449-454.





4. Gauen K., Dailey R., Laiman J., Zi Y. Comparison of Visual Datasets for Machine Learning // 2017 IEEE International Conference on Information Reuse and Integration (IRI), 2017, pp. 346-355.

5. Silberman N., Hoiem D., Kohli P., Fergus R. Indoor Segmentation and Support Inference from RGBD Images // Conference: European Conference on Computer Vision, 2012, pp. 746-760.

6. Wzorek P., Kryjak T. Training dataset generation for automatic registration of a duplicate bridge game // arXiv preprint arXiv: 2109.11.861v1, 2021.

7. Farou Z., Mouhoub N., Horvath T. Data Generation Using Gene Expression Generator // Synthetic Data Generation, 2020.

8. Barbosa I.B., Cristani M., Rognhaugen A., Theoharis T. Looking beyond Appearances: Synthetic Training Data for Deep CNNs in Re-identification // ArXiv preprint arXiv: 1701.03153v2, 2018.

9. Georgakis G., Mousavian A., Berg Alexander C., Kosecka J. Synthesizing Training Data for Object Detection in Indoor Scenes // ArXiv abs/1702.07836, 2017.

10. Chougule A., Pawar A., Kamble R., Mujawar J., Bhide A. Recognizing Fresh and Rotten Fruits Using Deep Learning Techniques // Data Engineering and Intelligent Computing, 2021, pp. 205-212.

### References

1. Solov'ev R.A., Tel'puhov D.V., Kustov A.G. Inzhenernyj vestnik Dona, 2017, №4. URL: [ivdon.ru/ru/magazine/archive/n4y2017/4433](http://ivdon.ru/ru/magazine/archive/n4y2017/4433).

2. Galushka V.V., Fathi V.A. Inzhenernyj vestnik Dona, 2013, №2. URL: [ivdon.ru/uploads/article/pdf/IVD\\_7\\_Galushka.pdf\\_1597.pdf](http://ivdon.ru/uploads/article/pdf/IVD_7_Galushka.pdf_1597.pdf).

3. Li C., Zhang Y., Qu, Y. 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI), 2018, pp. 449-454.

4. Gauen K., Dailey R., Laiman J., Zi Y. 2017 IEEE International Conference on Information Reuse and Integration (IRI), 2017, pp. 346-355.



5. Silberman N., Hoiem D., Kohli P., Fergus R. Conference: European Conference on Computer Vision, 2012, pp. 746-760.
6. Wzorek P., Kryjak T. arXiv preprint arXiv:2109.11.861v1, 2021.
7. Farou Z., Mouhoub N., Horvath T. Synthetic Data Generation, 2020.
8. Barbosa I.B., Cristani M., Rognhaugen A., Theoharis T. ArXiv preprint arXiv: 1701.03153v2, 2018.
9. Georgakis G., Mousavian A., Berg Alexander C., Kosecka J. ArXiv abs/1702.07836, 2017.
10. Chougule A., Pawar A., Kamble R., Mujawar J., Bhide Data Engineering and Intelligent Computing, 2021, pp. 205-212.