

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**  
**(ФГБОУ ВО «КубГУ»)**

**Факультет математики и компьютерных наук**  
**Кафедра математических и компьютерных методов**

## **КУРСОВАЯ РАБОТА**

### **РАЗРАБОТКА СИСТЕМЫ ГЕНЕРАЦИИ И ВЕРИФИКАЦИИ СИНТЕТИЧЕСКИХ ДАННЫХ ДЛЯ ЗАДАЧ КЛАССИФИКАЦИИ ИЗОБРАЖЕНИЙ**

Работу выполнила \_\_\_\_\_ О.К. Марчук  
(подпись)

Направление подготовки 02.03.01 Математика и компьютерные науки курс 3

Направленность (профиль) Математическое и компьютерное моделирование

Научный руководитель  
ст.преп \_\_\_\_\_ А.П. Невечеря  
(подпись, дата)

Нормоконтролер  
ст. лаборант \_\_\_\_\_ В.В. Писоцкая  
(подпись, дата)

Краснодар  
2025

## **ВВЕДЕНИЕ**

Сбор и разметка достаточного количества реальных данных часто связана с большими временными или материальными затратами. Чтобы добавить данных для обучения классификатора, используются генераторы, но изображения, выводимые этими генераторами, необходимо верифицировать - определить, сохраняются ли в них статистические свойства, характерные для реальных данных.

Объект исследования — влияние синтетических данных на эффективность классификатора изображений.

Предмет исследования — изучение классифицирующих способностей сверточных ИНС для классов дефектов на листах металла.

Цель работы: проверить гипотезу о том, что датасет, состоящий из фотографий листов металла можно дополнить или частично заменить синтетическим, что приведёт к возрастанию точности.

Задачи:

- разработка генератора изображений;
- разработка верификатора этих изображений;
- проведение А/В тестирования как метода верификации.

В ходе работы будут проведены следующие действия:

- анализ задачи и поиск информации о разработке аддонов на Blender;
- изучение архитектур свёрточных нейросетей;
- подтверждение или опровержение гипотезы о дополнении датасета синтетическими данными;

Требования к программе:

- имплементация нескольких методов разметки;
- получение метрик модели;
- наличие графического интерфейса, интегрированного в Blender;
- возможность настройки всех дефектов металла.

```

real_dataset_path, synthetic_dataset_path, epoch_count,
batch_count, test_size = sys.argv[1:] "total_time": -1}
reduce_lr = ReduceLROnPlateau(monitor='val_loss', factor=0.5,
patience=2, min_lr=1e-6)
early_stopping = EarlyStopping(monitor='val_loss', patience=5,
restore_best_weights=True)

X_synthetic_full, y_synthetic_full =
utils.load_dataset(synthetic_dataset_path)
X_real_full, y_real_full = utils.load_dataset(real_dataset_path)

for i in range(11):
    real_size = round(1 - i/10, 1)
    synthetic_size = round(i/10, 1)

    real_params = X_real_full, y_real_full, real_size
    synthetic_params = X_synthetic_full, y_synthetic_full,
synthetic_size

    X_mixed, y_mixed, X_test, y_test =
utils.get_mixed_data(real_params, synthetic_params, test_size)

    real_elements_num, synthetic_elements_num =
utils.get_number_of_elements(real_params, synthetic_params)
    model = build_segmentation_model()
    model.compile(optimizer=Adam(learning_rate=1e-3),
                  loss=utils.dice_loss,

```

Формула для dice\_loss, вычисляемая на основе истинных и предсказанных значений для класса пикселей:

$$loss = 1 - 2 \frac{\sum(y_{true} \cdot y_{pred})}{\sum(y_{true}) + \sum(y_{pred})}$$

```

metrics=[keras.metrics.BinaryIoU(target_class_ids=(0, 1),
threshold=0.5, name=None, dtype=None)])
    history = model.fit(X_mixed, y_mixed,
                          validation_data=(X_test, y_test),
                          epochs=epoch_count,
                          batch_size=batch_count,
                          verbose=1,
                          callbacks=[reduce_lr, early_stopping])

```

Так как класса всего два (есть дефект или нет дефекта), то используется метрика BinaryIoU — Intersection over Union. Чем выше количество перекрывающихся пикселей на предсказанной и на истинной битмаске, тем выше значение этой метрики.

## 2.3 Исследование результатов работы верификатора

Были проведены следующие эксперименты:

- дополнение датасета, состоящего из реальных данных, синтетическим датасетом в разных пропорциях;
- замена части реальных данных их синтетическими аналогами;
- подтверждение или опровержение гипотезы о дополнении датасета синтетическими данными.

При замене реальных данных синтетическими (90% реальных данных, 10% синтетических) наблюдалось консистентное незначительное повышение метрики Intersection over Union и более значительное понижение значения функции потерь (dice\_loss). При дальнейшем уменьшении количества реальных данных точность распознавания начинает падать ниже тех значений, которые наблюдались при 100% реального датасета и 0% синтетического.

Значения, выведенные в результате эксперимента с заменой (усреднённые данные трёх запусков программы), график ниже:

```
iou = [0.8536, 0.8776, 0.8732, 0.8596, 0.8465, 0.8221, 0.8328,  
0.7950, 0.7393, 0.7376, 0.6980]  
loss = [0.1923, 0.1660, 0.1734, 0.1886, 0.2009, 0.2353, 0.2209,  
0.2711, 0.3637, 0.3674, 0.4445]  
synthetic_size = [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8,  
0.9, 1.0]
```

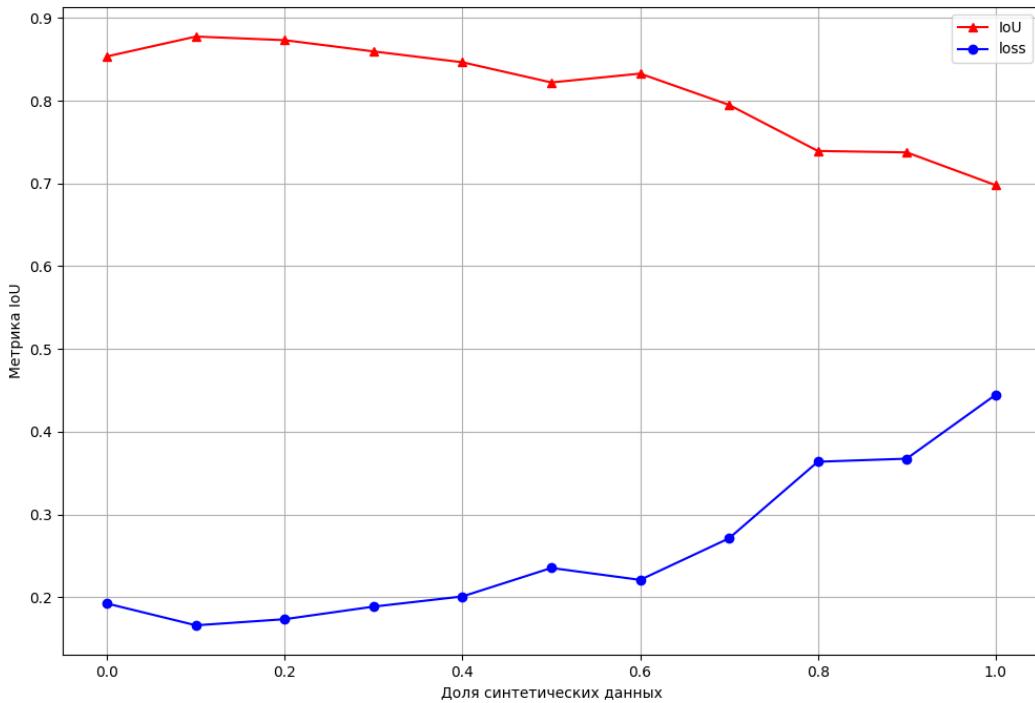


Рисунок 1 — График зависимости IoU от доли синтетических данных в тесте с заменой

График, полученный в ходе эксперимента с дополнением реального датасета синтетическими данными. При данной архитектуре достигается переобучение, и значительных изменений метрик в среднем не выявлено:

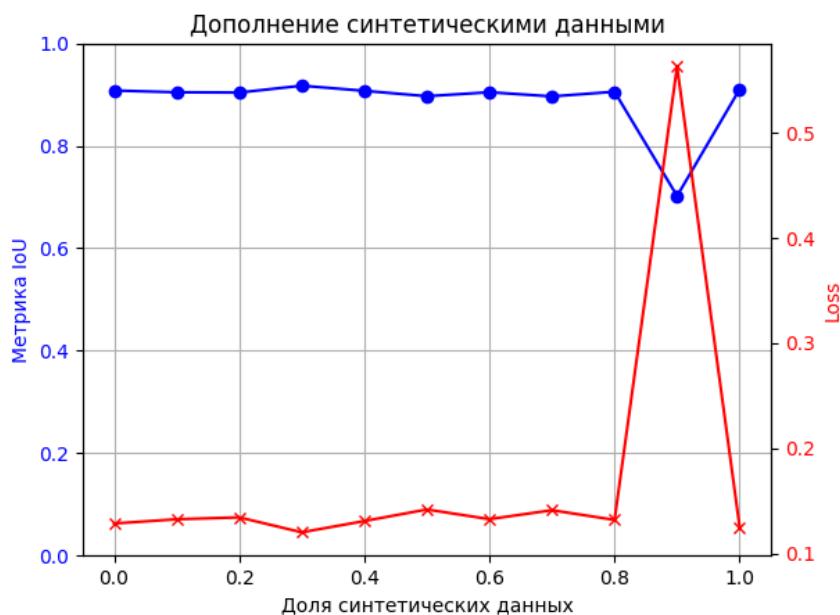


Рисунок 2 — График зависимости IoU от доли синтетических данных в тесте с дополнением

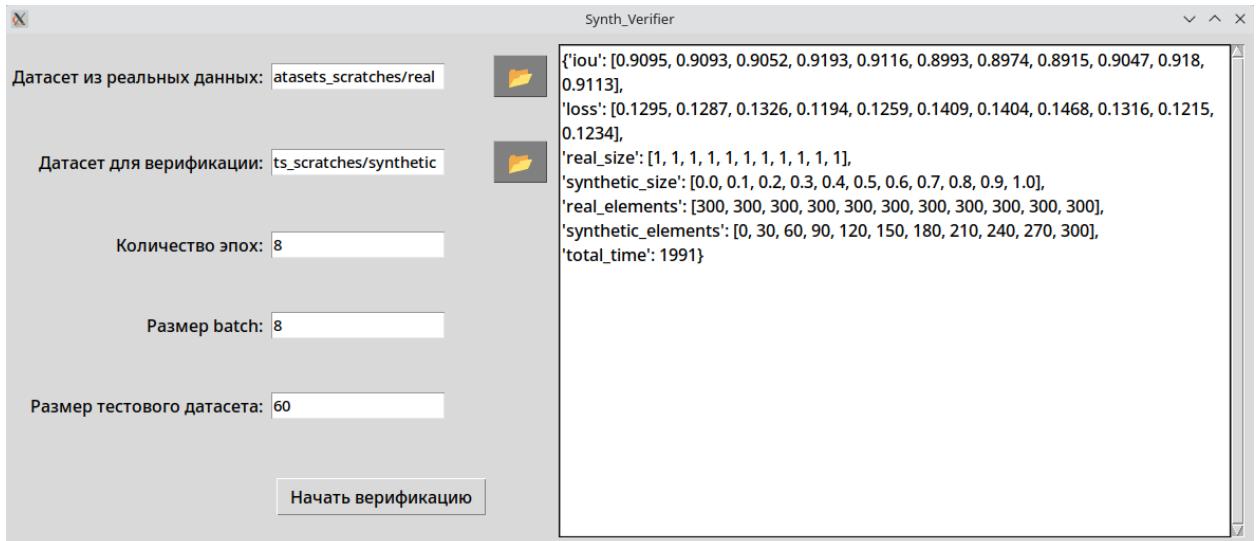


Рисунок 3 — Интерфейс верификатора

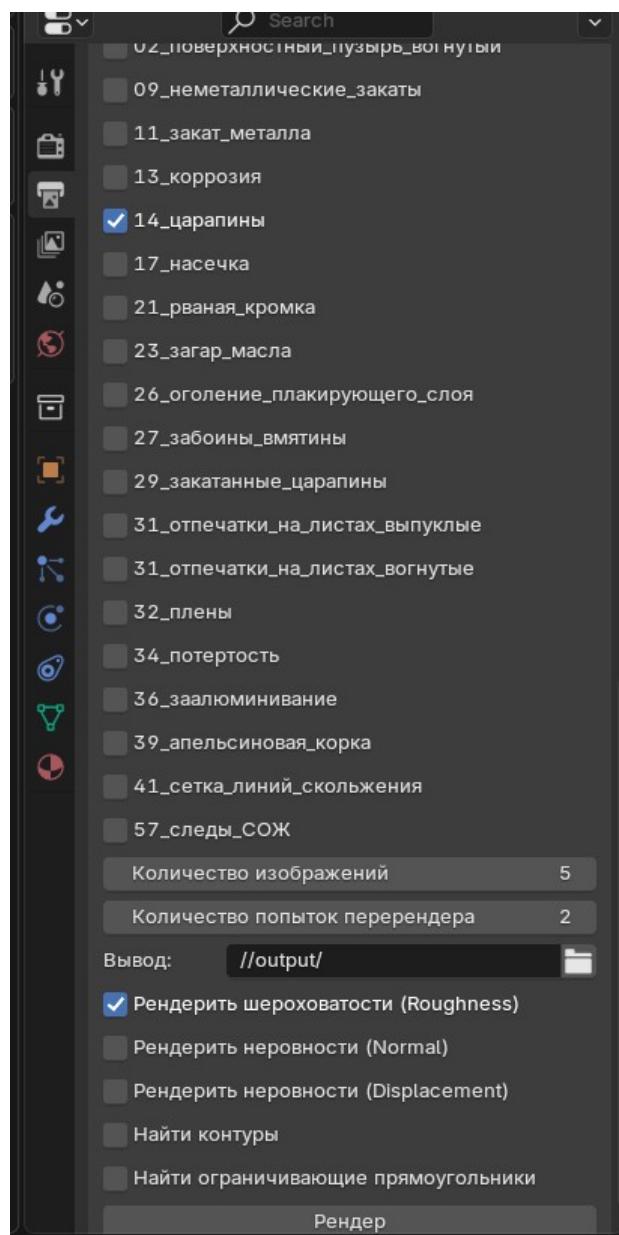


Рисунок 4 — Интерфейс генератора

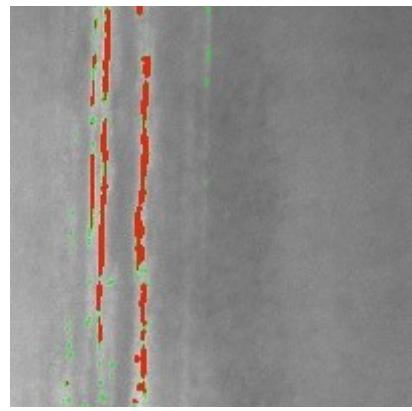


Рисунок 5 — Пример предсказанных дефектов

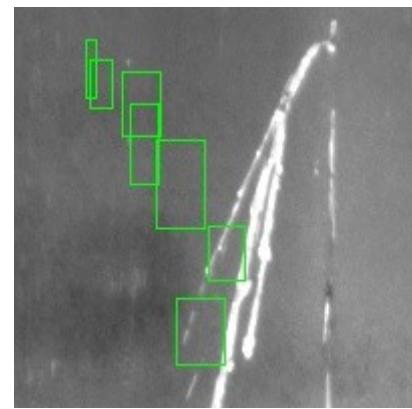


Рисунок 6 — Пример предсказанных дефектов до имплементации битмасок

## **ЗАКЛЮЧЕНИЕ**

В ходе работы выполнены следующие задачи:

- разработан генератор синтетических данных;
- разработан верификатор сгенерированного датасета;
- проведено А/В тестирование с использованием реального датасета и получены данные о достоверности синтетических данных.

Также можно доработать программу, добавив:

- автоматическую настройку параметров генерации для конкретного датасета;
- более точный метод оценки датасетов;
- поддержку распределённых вычислений для одновременного обучения нескольких итераций модели.

Перспективы развития: даже при незначительной настройке генератора пользователем можно добиться небольшого, но устойчивого повышения точности распознавания дефектов. Если существует такой алгоритм, который на вход принимает текущие значения метрик и выводит набор параметров для активных дефектов и окружения, то можно достичь ещё большего роста метрик.