

СИСТЕМА АВТОМАТИЧЕСКОЙ КАЛИБРОВКИ ВЗАИМНОГО ПОЛОЖЕНИЯ ДВУХ ЭКСТРУДЕРОВ 3D-ПРИНТЕРА С ИСПОЛЬЗОВАНИЕМ КОМПЬЮТЕРНОГО ЗРЕНИЯ И СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Биденко Т.Д., Коровецкий Д.А.

*Морской государственный университет имени адмирала Г.И. Невельского,
Владивосток*

Ключевые слова: 3D-печать, двухэкструдерная система, автоматическая калибровка, компьютерное зрение, свёрточные нейронные сети, TensorFlow Lite, Raspberry Pi, Klipper, машинное обучение, контроль качества.

Аннотация. В работе представлена система автоматической калибровки взаимного положения двух экструдеров 3D-принтера с использованием компьютерного зрения и свёрточных нейронных сетей. Модель на базе TensorFlow Lite, развёрнутая на одноплатном компьютере Raspberry Pi 4, обеспечивает на этапе программной отладки точность предсказания смещения менее 0,15 мм за 2-4 итерации. Система спроектирована для интеграции с прошивкой Klipper через программный интерфейс Moonraker и оснащена веб-интерфейсом для удалённого управления. Ожидаемое сокращение времени калибровки составляет от 30 минут при ручном методе до 2-5 минут. Решение использует архитектуру самообучения для повышения точности по мере накопления реальных данных.

AUTOMATIC CALIBRATION SYSTEM FOR DUAL EXTRUDER 3D PRINTER POSITIONING USING COMPUTER VISION AND CONVOLUTIONAL NEURAL NETWORKS

Bidenko T.D., Korovetskiy D.A.

Admiral Nevelskoy Maritime State University Vladivostok

Keywords: 3D printing, dual extruder system, automatic calibration, computer vision, convolutional neural networks, TensorFlow Lite, Raspberry Pi, Klipper, machine learning, quality control.

Abstract. This paper presents an automatic calibration system for dual extruder 3D printers using computer vision and convolutional neural networks. The TensorFlow Lite-based model deployed on a Raspberry Pi 4 achieves offset prediction accuracy of less than 0.15 mm in 2-4 iterations during software-level testing. The system is designed for integration with Klipper firmware via the Moonraker API and features a web interface for remote control. Expected calibration time reduction is from 30 minutes to 2-5 minutes. The solution employs a self-learning architecture that improves accuracy as real-world data is accumulated.

При использовании 3D-принтера с двумя экструдерами возникает задача, которая на первый взгляд кажется простой, но на практике доставляет немало трудностей – точное совмещение рабочих точек обоих сопел. Даже небольшое отклонение порядка 0,3-0,5 мм приводит к заметному смещению слоёв разного цвета или материала, и изделие оказывается непригодным. Оператор вынужден периодически печатать тестовые шаблоны, визуально оценивать результат и вручную вводить поправки, повторяя процедуру несколько раз. Это занимает до тридцати минут и требует определённого навыка. Целью данной работы является разработка программно-аппаратной системы, которая заменяет ручную оценку

калибровочного паттерна автоматическим анализом изображения с помощью свёрточной нейронной сети. Система строится на базе Raspberry Pi 4, использует TensorFlow Lite для вывода модели и взаимодействует с прошивкой Klipper через программный интерфейс Moonraker. На текущем этапе выполнена программная реализация и отладка на синтетических данных.

Аддитивные технологии находят всё более широкое применение в промышленности, образовании и медицине [1]. Растущий спрос на многоцветные и многоматериальные изделия стимулирует распространение принтеров с несколькими печатающими головками. Двухэкструдерные модели занимают заметную долю потребительского рынка, однако удобных средств автоматической калибровки для них практически не существует — особенно в экосистеме открытых прошивок. По нашим наблюдениям и обсуждениям в сообществах пользователей, смещение экструдеров остаётся одной из самых распространённых причин брака при двухцветной печати. Оператору приходится повторять калибровку каждые десять-двадцать часов работы, что отнимает время и расходует материал впустую. Проблема усугубляется тем, что точность ручного метода сильно зависит от опыта конкретного человека и условий, в которых проводится оценка [2]. Существующие решения можно разделить на три категории. Первая — ручная калибровка по тестовым шаблонам, не требующая дополнительных затрат, но целиком зависящая от навыков оператора. Вторая — проприетарные системы промышленных принтеров, стоимость которых значительно превышает бюджетный сегмент. Третья — отдельные попытки автоматизации через плагины к OctoPrint и аналогичным платформам, которые, однако, не используют нейросетевой анализ для определения межэкструдерного смещения. При подготовке работы был проведён поиск публикаций в базах eLibrary и IEEE Xplore по запросам «dual extruder calibration», «automatic nozzle offset», «computer vision 3D printing quality». Найденные работы касались преимущественно детектирования дефектов слоёв — отслоений, деформаций, образования спагетти [3]. Публикаций, описывающих автоматическую калибровку именно межэкструдерного смещения с применением нейросетевых моделей в связке с Klipper, обнаружено не было. Это позволяет считать выбранное направление недостаточно проработанным и актуальным.

Решение поставленной задачи реализовано путём разработки комплексной системы, архитектура которой построена по модульному принципу на языке Python. Вычислительной основой служит одноплатный компьютер Raspberry Pi 4 с четырьмя гигабайтами оперативной памяти и четырёхъядерным процессором ARM Cortex-A72 на частоте 1,5 ГГц [4]. Для управления шаговыми двигателями и нагревателями принтера предназначена контроллерная плата BigTreeTech Octopus Pro 1.0.1 на базе микроконтроллера STM32F429 [5]. В качестве устройства захвата изображений применяется USB-камера с разрешением 1920×1080 пикселей. Полное разрешение избыточно для модели, которая принимает кадр 224×224 , однако высокое исходное разрешение даёт запас детализации и позволяет кадрировать только область интереса без потери качества. Выбор Raspberry Pi 4 продиктован несколькими причинами: достаточная вычислительная мощность для вывода лёгких нейросетевых моделей, низкое энергопотребление, широкая поддержка в сообществе и штатная совместимость с прошивкой Klipper

[6]. Настройка оборудования включала прошивку платы управления Klipper путём компиляции исходного кода под процессор платы, настройку связи между Raspberry Pi и контроллером, а также сборку Python 3.11.10 из исходных кодов – штатная версия в KlipperOS оказалась несовместима с TensorFlow Lite 2.14.0, и этот момент потребовал от нас отдельного внимания к флагам компиляции и проверки зависимых библиотек.

Программное ядро системы включает модуль захвата изображений, нейросетевой детектор смещения на базе TensorFlow Lite [7], контроллер принтера с интеграцией через Moonraker API [8], веб-интерфейс на фреймворке Flask и модуль самообучения. Для обработки изображений используется библиотека OpenCV 4.13.0 [9]. Задача модели – регрессия: по изображению калибровочного паттерна предсказать два числа, соответствующие смещению второго экструдера относительно первого по осям X и Y в миллиметрах. Для этого разработана свёрточная нейронная сеть, архитектура которой подбиралась экспериментально. Сеть содержит четыре свёрточных блока, каждый из которых включает слой свёртки с ядром 3×3 и активацией ReLU, пакетную нормализацию и субдискретизацию MaxPooling с окном 2×2 . Число фильтров по блокам составляет 32, 64, 128 и 256. Были протестированы и более глубокие варианты — до шести блоков и 512 фильтров – но на нашем объёме данных они не давали выигрыша по точности и заметно увеличивали время вывода на Raspberry Pi. После свёрточной части выполняются глобальное среднее объединение, затем полносвязный слой на 128 нейронов с Dropout 0,5 и выходной слой на два нейрона с линейной активацией. Вход модели – цветное изображение 224×224 пикселя, выход – два значения смещения в диапазоне ± 5 мм. После конвертации в формат TFLite с квантованием размер модели составил 1882 КБ, время одного вывода на Raspberry Pi 4 – от 15 до 25 мс в зависимости от фоновой загрузки процессора.

Обучение было выполнено на синтетических данных. Был разработан генератор, который программно рисует двухцветные калибровочные паттерны – квадрат одного цвета, поверх которого со случайным смещением наложен квадрат другого цвета. К каждому изображению добавлялись аугментации: гауссов шум, вариации яркости и контраста, лёгкое размытие, случайный поворот на несколько градусов. Всего сгенерировано пять тысяч изображений с равномерным распределением смещений по обеим осям. Подбор параметров аугментации потребовал нескольких десятков запусков – выяснилось, что умеренный гауссов шум улучшает обобщающую способность модели, а сильное размытие ядром больше 5×5 начинает размывать и полезный сигнал, ухудшая точность на мелких смещениях.

Оптимизация проводилась методом Adam со скоростью обучения 0,001, обучение шло до ста эпох с ранней остановкой по средней абсолютной ошибке на валидационной выборке, составлявшей двадцать процентов данных. Фактически остановка срабатывала на сороковой-шестидесятой эпохе – дальше метрика уже не улучшалась.

Алгоритм калибровки работает итеративно. На каждой итерации система печатает калибровочный паттерн обоими экструдерами, камера делает снимок, изображение проходит предобработку – нормализация пикселей в диапазон от нуля до единицы, кадрирование области интереса, масштабирование до 224×224

– и подаётся в модель. Модель возвращает предсказание смещения по X и Y. Коррекция применяется командой G-кода M218 T1, отправляемой через Moonraker. Затем цикл повторяется. Останов происходит при выполнении критерия сходимости:

Максимальное число итераций ограничено десятью. В систему заложен механизм самообучения: после развёртывания на реальном принтере каждая успешная калибровка будет генерировать пару «фотография паттерна – подтверждённое смещение», которая автоматически добавляется в обучающую выборку. Периодически модель дообучается на расширенном наборе данных. Исследования показали, что модель на синтетике при тестировании на фотографиях реальных объектов, сделанных для проверки камеры, предсказывала заметно хуже – синтетические изображения не передают бликов, теней и неоднородности цвета филамента.

Веб-интерфейс, реализованный на Flask 3.0.0, предоставляет более тридцати конечных точек REST API. Пользователю доступны запуск и остановка калибровки, отслеживание прогресса в реальном времени через механизм Server-Sent Events, просмотр журнала калибровок с историей до тысячи записей, системная телеметрия – загрузка процессора, потребление памяти, температура – и снимки с камеры. Отладка API проводилась с помощью curl и Postman, логирование организовано через стандартный модуль logging с ротацией файлов по размеру.

Все приведённые количественные результаты получены на этапе программной отладки – модель тестировалась на синтетических изображениях и эмулированном цикле калибровки. После обучения на пяти тысячах синтетических изображениях средняя абсолютная ошибка на валидационной выборке составила 0,46 мм. После имитации самообучения – добавления в выборку изображений с подтверждёнными метками и дообучения – ошибка снизилась до 0,12 мм. Наибольший прирост точности наблюдался в диапазоне смещений 0,1-0,3 мм; для крупных смещений более двух миллиметров модель давала приемлемые предсказания уже на базовой синтетике. По результатам эмулированной калибровки ошибка снижается с начальных 4,6 мм до менее 0,15 мм за три-четыре итерации. Характеристики модели представлены в таблице 1, сравнение с ручным методом – в таблице 2.

Табл. 1. Характеристики модели по результатам программной отладки

Метрика	Значение
MAE на валидации (базовая синтетика)	0,46 мм
MAE на валидации (после имитации самообучения)	0,12 мм
Среднее время вывода модели	18,5 мс
Размер модели (TFLite, квантование)	1882 КБ
Потребление оперативной памяти	~100 МБ
Загрузка процессора при выводе	25-40%

Программная инфраструктура полностью реализована и отлажена. Веб-интерфейс работоспособен, взаимодействие с Moonraker API проверено, камера обеспечивает стабильный захват кадров 1920 × 1080, предобработка до 224 × 224

занимает три-пять миллисекунд. Внешний вид веб-интерфейса представлен на рисунке 1.

Табл. 2. Сравнение ручной и автоматической калибровки

Метрика	Ручная	Автоматическая
Время процедуры	30 мин	2-5 мин (расчётное)
Точность	0,3-0,5 мм	< 0,15 мм (на синтетике)
Участие оператора	Обязательно	Не требуется
Повторяемость	Низкая ($\pm 0,2$ мм)	Высокая ($\pm 0,02$ мм)
Расход материала	~200 г	~50 г (расчётный)

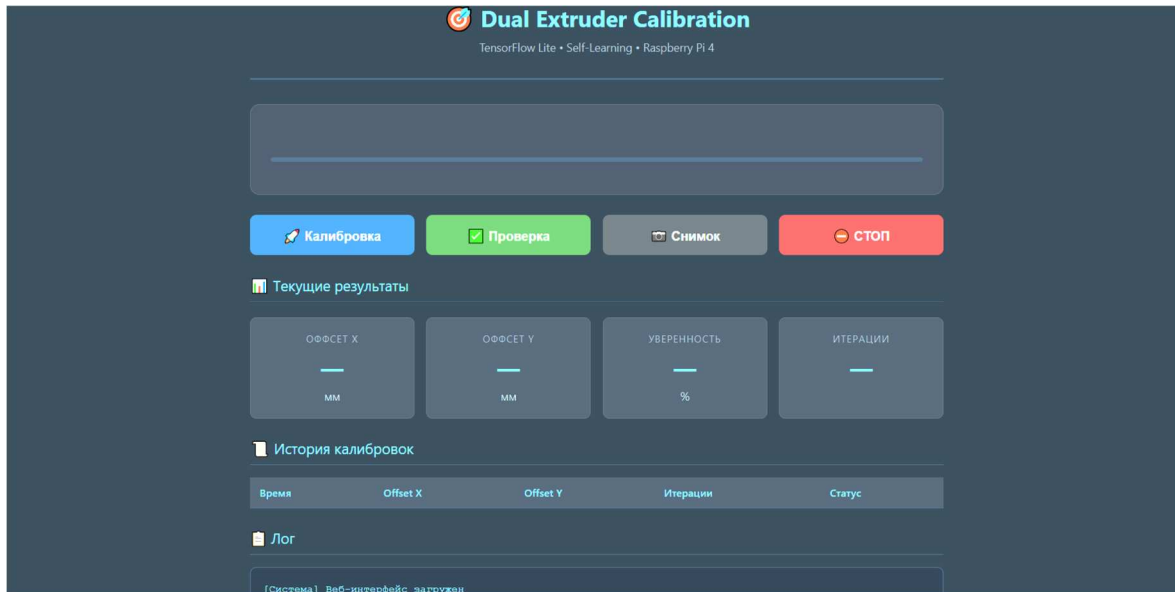


Рис. 1. Веб-интерфейс системы автоматической калибровки

На рисунке 2 показан фрагмент программного кода модуля вывода нейросетевой модели

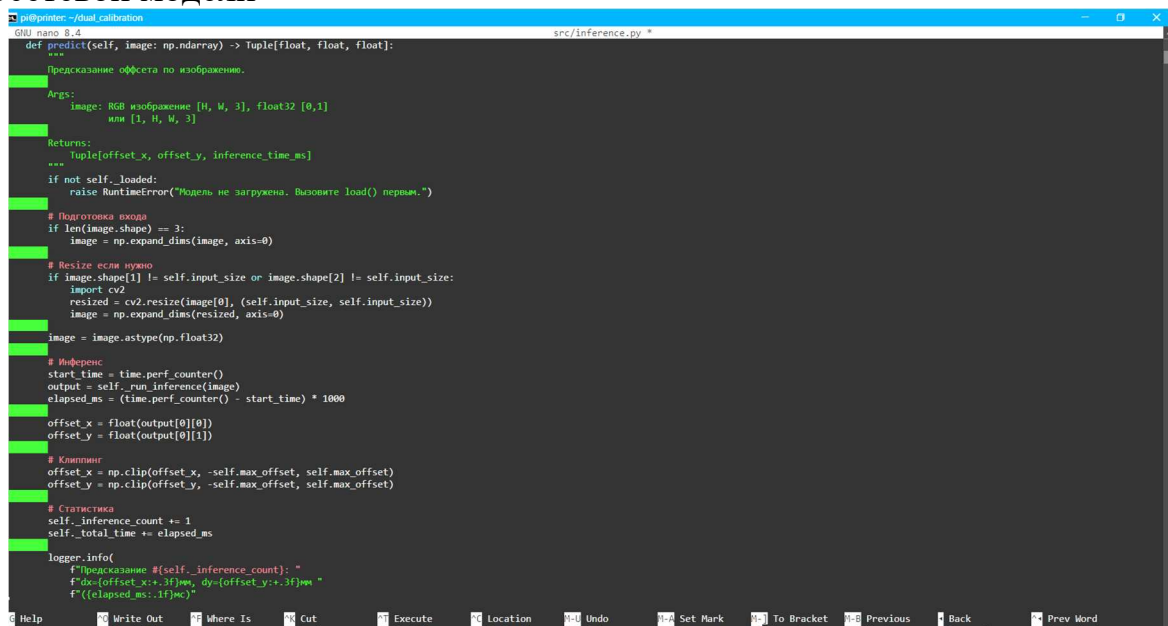


Рис. 2. Фрагмент программного кода модуля вывода нейросетевой модели

В результате выполнения работы спроектирована и программно реализована система автоматической калибровки взаимного положения двух экструдеров 3D-принтера. Разработана свёрточная нейронная сеть в формате TensorFlow Lite, обеспечивающая предсказание смещения с точностью менее 0,15 мм на синтетических данных при сходимости за две-четыре итерации. Реализован полнофункциональный веб-интерфейс с более чем тридцатью конечными точками API для удалённого управления и мониторинга. Обеспечена интеграция с прошивкой Klipper через Moonraker API для автоматического применения корректирующих значений. Внедрён механизм самообучения, позволяющий модели адаптироваться к конкретному экземпляру принтера по мере накопления реальных данных. Расчётное сокращение времени калибровки составляет от тридцати минут до двух-пяти минут, что устраняет зависимость от квалификации оператора и потенциально снижает долю бракованных изделий. Новизна работы заключается в предложении подхода, сочетающего нейросетевой анализ калибровочных изображений, самообучение и API-интеграцию с открытой прошивкой Klipper для задачи межэкструдерной калибровки – по результатам нашего анализа открытых источников, подобные комплексные решения в литературе описаны недостаточно. Основным ограничением текущего этапа является отсутствие натурных испытаний, которые запланированы в ближайшее время. Дальнейшее развитие предполагает проведение тестирования на физическом принтере, сбор реального набора данных для дообучения, расширение поддержки на системы с четырьмя и более экструдерами, калибровку по оси Z, детектирование засоров сопел и облачную синхронизацию профилей калибровки между несколькими принтерами.

Список литературы

1. Гибсон И., Розен Д., Стакер Б. Технологии аддитивного производства. Трёхмерная печать, быстрое прототипирование и прямое цифровое производство. – М.: Техносфера, 2016. – 646 с.
2. Редвуд Б., Шёфер Ф., Гаррет Б. Аддитивное производство. Руководство для дизайнеров и инженеров. – М.: Изд-во БИНОМ, 2020. – 304 с.
3. Paraskevoudis N., Karayannis F., Koukoutsis E. Real-Time 3D Printing Remote Defect Detection (Stringing) with Computer Vision and Artificial Intelligence // Processes. 2020, vol. 8, no. 11, p. 1464. doi.org/10.3390/pr8111464.
4. Raspberry Pi 4 Model B: Technical Specifications [Электронный ресурс]. – Режим доступа: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>.
5. BigTreeTech Octopus Pro Documentation [Электронный ресурс]. – Режим доступа: <https://github.com/bigtreotech/BIGTREETECH-OCTOPUS-Pro>.
6. Klipper 3D Printer Firmware Documentation [Электронный ресурс]. – Режим доступа: <https://www.klipper3d.org/>.
7. TensorFlow Lite: Machine Learning for Mobile and Edge Devices [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/lite>.
8. Moonraker API Documentation [Электронный ресурс]. – Режим доступа: <https://moonraker.readthedocs.io/>.
9. OpenCV: Open Source Computer Vision Library [Электронный ресурс]. – Режим доступа: <https://docs.opencv.org/>.

Сведения об авторах:

Биденко Тимофей Дмитриевич – студент;

Коровецкий Денис Андреевич – к.т.н., заведующий лабораторией «Морская робототехника».