

Салем, делегаты!

Каждый строительный проект, от моста до нефтеперерабатывающего завода, может включать в себя тысячи документов, разрешений и чертежей. Наша миссия - использовать ИИ для чтения и понимания этих документов, гарантируя, что проекты безопасны, законны и соответствуют всем нормам.

Это огромная и сложная задача, и сейчас мы приглашаем вас помочь нам решить одну из ее важных частей.

Задача: Создать "Цифрового Инспектора"

Проверка сложных документов - это медленная ручная работа. Эксперт может потратить множество часов, просто чтобы убедиться, что все базовые проверки выполнены. Мы хотим, чтобы вы автоматизировали "рутинную" часть этой инспекции.

Ваша миссия — создать инструмент Компьютерного Зрения (Computer Vision), который сможет автоматически находить и отмечать три ключевых элемента на любом строительном документе:

Подписи

Печати / Штампы

QR-коды (которые часто ведут на лицензии или нормативную информацию)

По сути, это задача по обнаружению объектов (object detection), которую нужно решить в сжатые сроки. Ваш инструмент будет действовать как "Цифровой Инспектор", сканируя документ и мгновенно указывая:

"Вот подпись, вот печать, а вот QR-код".

Что мы предоставляем

Датасет: Специальный тестовый набор примеров реальных строительных документов.

Проблему: Мы говорим вам, *что* искать.

Полную свободу: Вы решаете, *как* искать. Используйте любые инструменты – Python, TensorFlow, PyTorch, OpenCV, YOLO, всё разрешено.

Требования к сдаче работы

У вас есть 24 часа. К дедлайну вы должны предоставить:

01

Презентацию (визуализацию): Цифровую презентацию (например, PowerPoint, Google Slides), где вы объясняете ваш подход, использованные методы, сложности и результаты.

02

Видео: Короткое видео (до 3 минут), демонстрирующее ваше решение и рассказывающее о презентации.

03

Код (видимость): Доступ к вашему коду (предпочтительно — репозиторий на GitHub). Также приложите **README.md**, где описано, как запустить код, какие пакеты установить или где посмотреть демо. А также **REQUIREMENTS.TXT**.

Важные детали

- ❏ Важно: Ваше решение должно уметь выдавать визуальный результат. Не нужно писать скрипты, которые просто печатают координаты в консоль. Ваш проект должен явно демонстрировать, что он нашел!

Хорошие примеры

- Визуализировать ограничивающие рамки (bounding boxes) на изображении, показывая найденные подписи, печати и QR-коды.
- Веб-приложение, где можно загрузить файл и увидеть результаты.
- Визуализация в Matplotlib.

Итоговый результат должен быть представлен в формате JSON, что позволит легко верифицировать данные и использовать их для визуальной демонстрации во время питчинга.

Как вас будут оценивать

Мы ищем лучшее комплексное решение. Работы будут оцениваться по пяти ключевым критериям:

- 1 Точность и надёжность:** Насколько корректно и стабильно ваше решение работает с разными типами документов?
 - 2 Скорость и оптимизация:** Насколько решение эффективно, масштабируемо и устойчиво к ошибкам? Справится ли оно с обработкой 1000 документов?
 - 3 Техническая сложность:** Насколько инновационен ваш подход? Использовали ли вы готовую модель или улучшили (fine-tuned) её? Ваш код чистый, структурированный и устойчивый?
 - 4 Презентация:** Насколько понятно вы объяснили, что сделали, почему и как это работает? Даже отличная модель без ясного объяснения не победит.
 - 5 Видение:** Как вы видите развитие и масштабирование вашего решения в будущем — при большем объёме данных или вычислительных ресурсов?
-