



Санкт-Петербургский государственный университет
Математическое обеспечение и администрирование информационных систем

Реализация алгоритма сопоставления результатов детекции торцов брёвен на изображениях штабелей круглого леса

Максим Алексеевич Паршин
18.Б09-мм

Научный руководитель: старший преподаватель кафедры системного программирования
Смирнов М. Н.

Консультант: менеджер проектов ООО «Системы Компьютерного Зрения» Ткачёва Д. А.

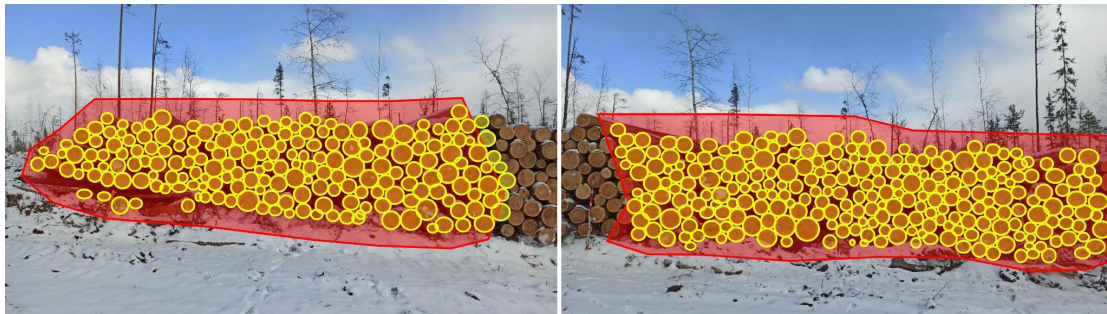
Санкт-Петербург
2021

Введение

- Автоматизация измерений объема круглых лесоматериалов в штабелях и на лесовозах при помощи Android-приложения
- Проект компании ООО «Системы Компьютерного Зрения»
- Распознавание торцов брёвен на фотографии
- Штабели могут быть длинными, одной фотографии недостаточно
- На последовательных фотографиях частей штабеля присутствуют дублирующиеся торцы, которые необходимо отбрасывать при подсчёте

Цель работы

- Для удаления дублирующих торцов используется область интереса, выделенная пользователем
- На левом изображении сохраняются торцы, пересекающиеся с областью, на правом — целиком лежащие в области
- **Проблема:** на практике пользователи не выделяют область точно, и это не всегда возможно
- **Цель работы** — уменьшить число дважды включенных/исключенных торцов при ручном выделении области



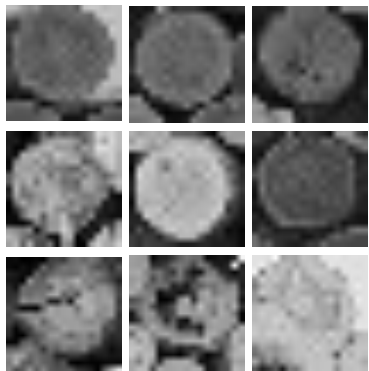
Постановка задачи

- Разработать модификацию текущего алгоритма исключения дубликатов торцов, использующую автоматическое сопоставление результатов детекции между соседними изображениями
- Реализовать модифицированную версию алгоритма
- Отобрать и подготовить данные для тестирования (последовательные фотографии частей штабеля с установленными вручную сопоставлениями)
- Провести тестирование и соответствующую доработку алгоритма
- Подготовить итоговую версию алгоритма к интеграции в существующее мобильное приложение для подсчёта древесины

Обзор

- **SIFT**

- Детектор особых точек на изображении
- Изображение как функция
- Направления и величины градиента в точках
- Инвариантен относительно поворота
- В данном случае строится не в найденной детектором особой точке, а в квадратном сегменте изображения, содержащем торец бревна



- **JavaCV**

- Java-интерфейс для библиотеки OpenCV
- Использование методов OpenCV в коде Android-приложения на Kotlin

Схема работы алгоритма

1. Выделение торцов, лежащих рядом с границей пользовательской области (рис. 1)
2. Поиск сопоставлений между выделенными торцами на левом и правом изображениях (рис. 2)
3. Добавление торцов, для которых найдено сопоставление, в пользовательскую область на левом изображении и их удаление из области на правом (рис. 3)



рис. 1

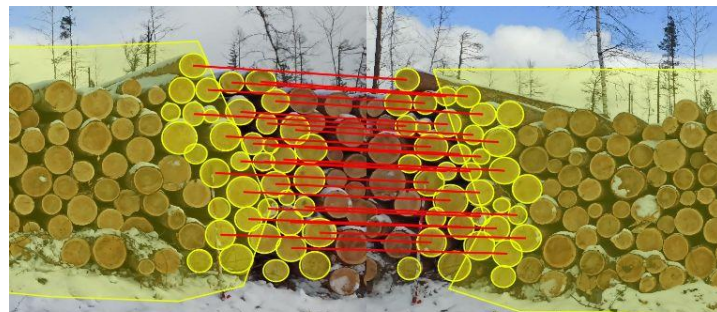


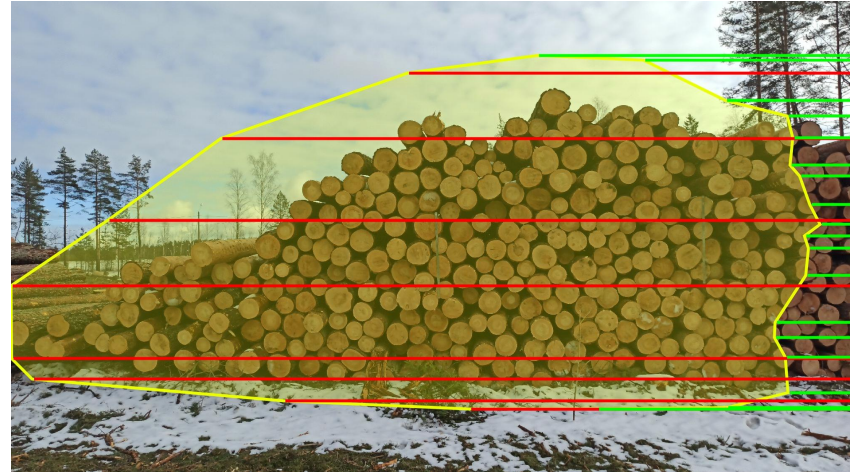
рис. 2



рис. 3

Выделение торцов, лежащих рядом с границей пользовательской области

1. Из вершин многоугольника области строятся горизонтальные лучи, направленные в сторону стыка
2. Рассматриваются те вершины многоугольника, лучи из которых не пересекают его
3. Выделяются торцы, центры которых лежат близко к сторонам многоугольника, соединяющим такие вершины



Поиск сопоставлений

- Осуществляется построение дескрипторов для всех торцов
- Каждому торцу слева сопоставляются к торцов справа с наиболее близкими дескрипторами (рис. 1)
- Отбрасываются сопоставления пар с высокой разницей относительных у-координат центров (рис. 2)

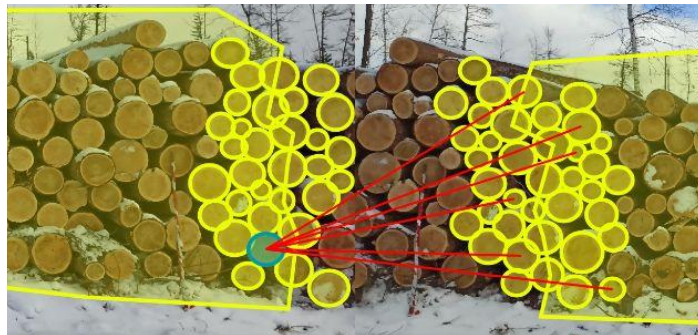


рис. 1

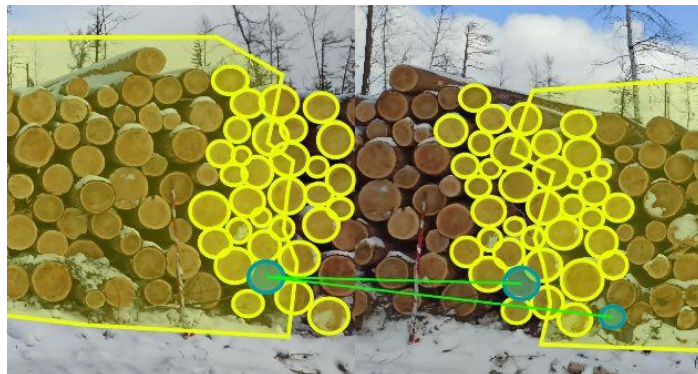


рис. 2

Поиск сопоставлений

- Для элементов каждой сопоставленной пары рассматриваются n соседних торцов вокруг (рис. 1)
- Отбрасываются сопоставления, для которых малое число соседей слева было сопоставлено соседям справа (рис. 2)
- Из оставшихся сопоставлений выбирается сопоставление с минимальным расстоянием между дескрипторами (рис. 3)

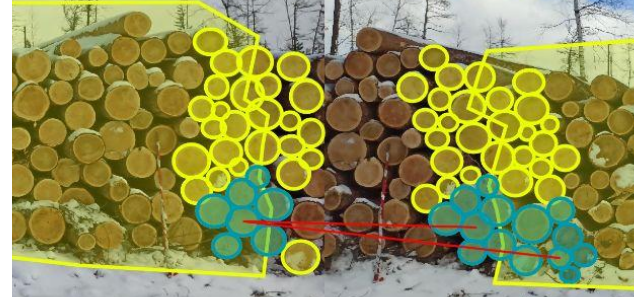


рис. 1

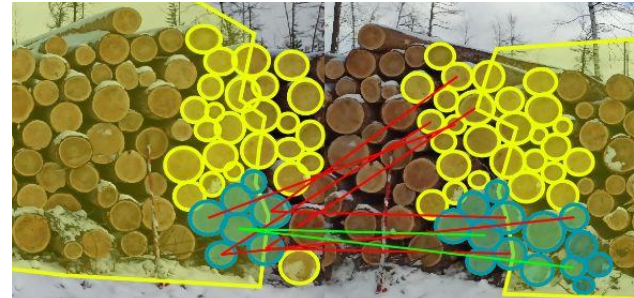


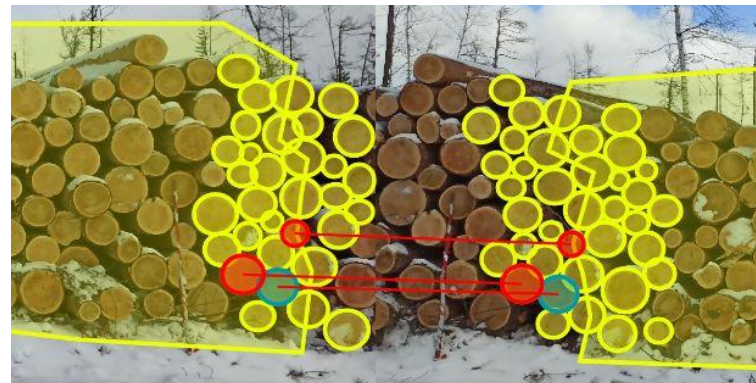
рис. 2



рис. 3

Поиск сопоставлений

- Строится обратное сопоставление и выбирается сопоставление с минимальным расстоянием между дескрипторами, чтобы получить сопоставление «один к одному»
- Дополнительно добавляются сопоставления между еще несопоставленными соседями, лежащими под сходным углом (рис.)



Тестирование алгоритма

- Метрика оценки результатов сопоставления торцов (второго шага алгоритма):

$$precision = \frac{TP}{TP + FP} \quad recall = \frac{TP}{TP + FN}$$

TP — число верно найденных сопоставлений

FP — число неверно найденных сопоставлений

FN — число ненайденных сопоставлений

- Метрика оценки результатов работы алгоритма:

$$SMC = \frac{S_{IE} + S_{EI}}{2S}$$

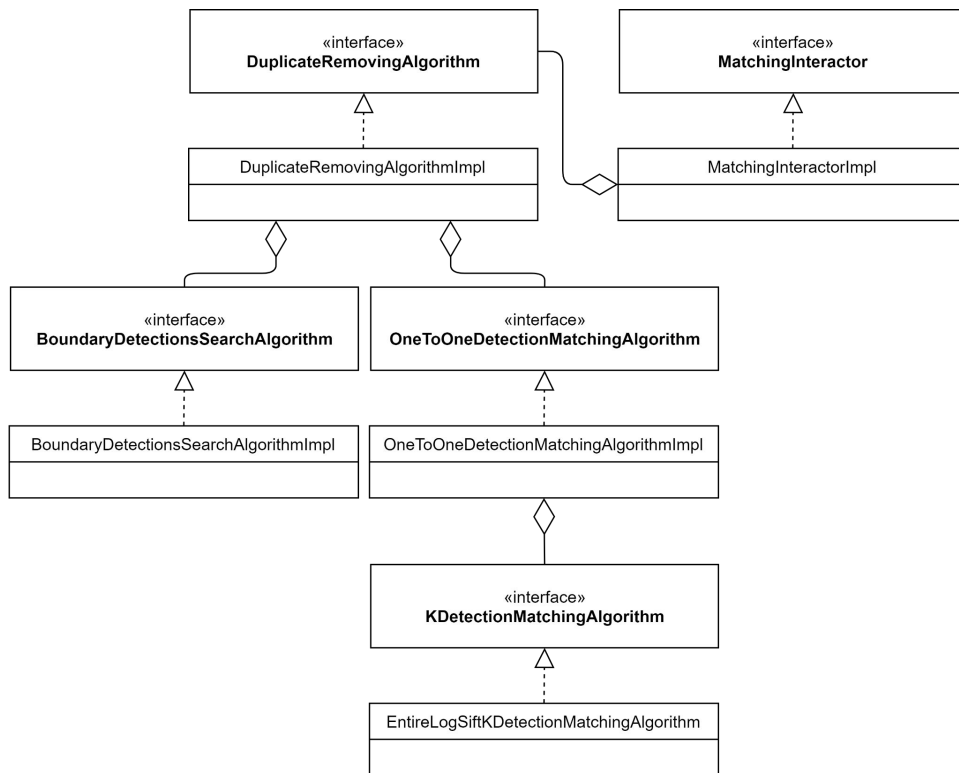
S_{IE} — число торцов, включенных в область слева и исключенных из области справа

S_{EI} — число торцов, исключенных из области слева и включенных в область справа

S — число торцов, присутствующих и на левом, и на правом изображении

набор входных данных	precision	recall	SMC, текущий алгоритм	SMC, улучшенный алгоритм	прирост SMC
1	1,000	0,828	1,000	1,000	0,000
2	1,000	0,839	1,000	1,000	0,000
3	0,942	0,733	0,942	0,962	0,019
4	0,962	0,774	0,962	1,000	0,038
5	0,827	0,786	0,827	0,981	0,154
6	1,000	0,706	1,000	1,000	0,000
7	0,981	0,933	0,981	1,000	0,019
8	0,975	0,633	0,975	0,983	0,008
9	0,983	0,787	0,983	0,992	0,008
10	0,916	0,653	0,916	0,966	0,050
11	0,958	0,776	0,958	0,983	0,025
12	0,916	0,620	0,916	0,958	0,042
13	0,916	0,650	0,916	0,983	0,067
14	0,933	0,773	0,933	0,966	0,034
15	0,986	0,795	0,986	0,986	0,000
16	0,986	0,676	0,986	0,986	0,000
17	0,945	0,677	0,945	0,973	0,027
18	0,945	0,868	0,945	1,000	0,055
19	0,986	0,744	0,986	0,986	0,000
20	0,836	0,939	0,836	0,986	0,151
21	0,978	0,585	0,978	0,978	0,000
22	0,972	0,615	0,972	0,978	0,006
23	0,905	0,586	0,905	0,961	0,056
24	0,927	0,662	0,927	0,939	0,011
25	0,939	0,594	0,939	0,972	0,034
26	0,950	0,688	0,950	0,961	0,011
27	0,950	0,658	0,950	0,966	0,017

Архитектура реализации



Результаты

- Разработана модификация текущего алгоритма исключения дубликатов торцов, использующая автоматическое сопоставление результатов детекции между соседними изображениями
- Реализована модифицированная версия алгоритма
- Отобраны и подготовлены данные для тестирования (последовательные фотографии частей штабеля с установленными вручную сопоставлениями)
- Проведено тестирование и соответствующая доработка алгоритма
- Итоговая версия алгоритма подготовлена к интеграции в существующее мобильное приложение для подсчёта древесины