Во время разработки метода распознавания были рассмотрены 51 класс возможных букв.

Т.е.:

1. Все строчные буквы русского алфавита за исключением буквы ‘ё’ (32 шт.)
2. Все цифры (10 шт.)
3. Прописные буквы русского алфавита, которые значимо не совпадают визуально чсо строчными – ‘А’, ‘Б’, ‘В’, ‘Г’, ‘Д’, ‘П’, ‘Р’, ‘T’, ‘У’. (9 шт.)

Таким образом, для каждого обработанного изображения буквы получалась какая-то вероятность того, что она является тем или иным классом (В сумме эти вероятности дают единицу).

Обозначим данный вектор вероятностей как p[i].

Обозначим количество букв в слове как n, тогда у нас есть n векторов-вероятностей для каждой буквы из данного слова.

Тогда для одного слова обозначим за arr[i][j] – двумерный массив вероятностей размерности [n x 51], который отображает, что i-ая буква данного слова с вероятностью arr[i][j] является j-ой буквой в рассматриваемых классах.

Теперь задача распознавания сводится к поиску ближайшей точки в 51-мерном пространстве к данной, где точкам в пространстве соответствует слово из словаря.

Чтобы более точно распознавать слова, можно использоваться следующий подход:

1. Сохранить словарь всех (или большинства наиболее встречаемых) слов русского языка
2. Для каждого слова из словаря длинны n посчитать степень «похожести» с помощью какой-либо метрики, для данных n векторов-вероятностей, полученных при распознавании.
3. Слово с наибольшей степенью похожести определить как результат распознавания.

Таким образом можно значительно увеличить точность метода распознавания.

Вторая часть УИРа будет рассматривать процесс разработки и создания требований, архитектуры и программной реализации android приложения для метода распознавания, используя клиент-серверное взаимодействие, разработанное ранее на этапе практики.

Таким образом можно рассмотреть следующие этапы как основные пункты для задания УИРа:

Аналитическая часть:

1. Изучение и сравнительный анализ методов увеличения точности метода распознавания текста.
2. Изучение и сравнительный анализ методов нечёткого поиска
3. Изучение существующих методологий и способов проектирования android приложения.
4. Оформление расширенного содержания пояснительной записки (РСПЗ)

Теоретическая часть:

1. Выбор алгоритмов увеличения точности распознавания.
2. Выбор методологии для проектировки android приложения

Инженерная часть:

1. Выбор языка программирования и архитектуры для реализации android-приложения
2. Программная реализация метода оптимизации и android-приложения

Технологическая и практическая часть:

1. Тестирование разработанного приложения
2. Выводы по результатам тестирования
3. Оформление пояснительной записки (ПЗ) и иллюстративного материала для доклада.