

Московский государственный университет имени М. В. Ломоносова



Факультет Вычислительной математики и Кибернетики
Кафедра Математических Методов Прогнозирования

Обработка и методы распознавания изображений

Отчет по лабораторной работе №2

Выполнил:
студент 3 курса 317 группы
Таскынов Ануар

Москва, 2016

Содержание.

1	Постановка задания.	2
2	Описание решения.	3
2.1	Бинаризация.	3
2.2	Выделение и фильтрация дефектов выпуклой оболочки. . .	4
2.3	Поиск большого пальца.	5
3	Кластеризация и результаты.	7

1 Постановка задания.

В данном задании необходимо было разработать программу для классификации изображений ладоней, по сгенерированным признакам. В качестве признакового описания строилась «линия пальцев», соединяющая основания пальцев с их кончиками.

Язык программирования Python 2. В качестве сторонней библиотеки использовалась библиотека компьютерного зрения OpenCV. Эксперименты приведены в Ipython Notebook'e, функции описаны в отдельном python-модуле. Выполнен уровень Expert.

В уровне Expert помимо генерации признаков, необходимо было провести кластеризацию изображений.

2 Описание решения.

2.1 Бинаризация.

Первым делом необходимо бинаризовать изображение. Это делалось следующим образом:

- Проводилась бинаризация по методу Оцу.
- Найденный порог t умножался на 1.6. Это было сделано для того, чтобы полностью отделить руку от фона, так как некоторые изображения при пороге t отделялись не полностью. Примеры показаны на рис. 1
- Далее удаляются мелкие объекты.



Рис. 1: Изображения, на которых плохо отработал порог t .

При умножении порога на 1.6 бинаризация значительно улучшается. Пример отработанной бинаризации на тех же изображениях показан на рис. 2



Изображение 012



Изображение 013

Рис. 2: Изображения, на которых плохо отработал порог $1.6t$.

2.2 Выделение и фильтрация дефектов выпуклой оболочки.

Дальше пойдет этап выделения дефектов выпуклой оболочки. Дефекты будут показывать на точки, куда было добавлено точек для создания выпуклой оболочки. Далее за *start*, *far*, *end* будут обозначаться начало дефекта, сам дефект, конец дефекта.

- Ищутся дефекты выпуклой оболочки.
- Ищется угол между точками *start*, *far* и *end*. Так как угол между пальцами в основном острый, брался угол равный 95 для отсеечения ненужных дефектов.
- Также были замечены мелкие дефекты, которые также имеют острый угол. Для их отсеечения считалась площадь треугольника на этих трех точках.

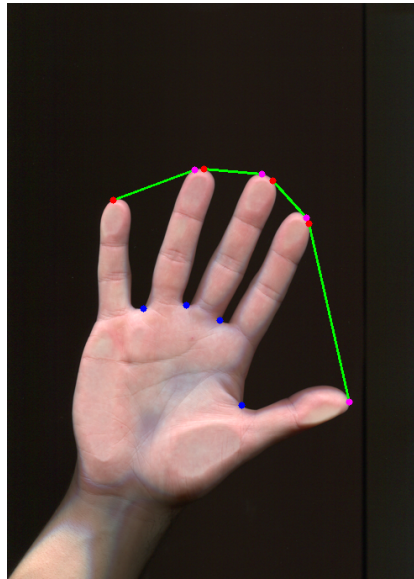


Рис. 3: Начало дефекта указано красным цветом, сам дефект синим, конец дефекта розовым.

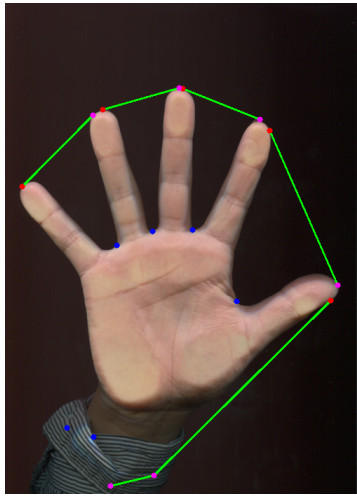
В результате проделанных манипуляций оставалось крайне мало дефектов не похожих на кончики пальцев и их основания. Но этого было не достаточно для полного отсечения ненужных дефектов (рис. 4). Для их фильтрации использовалось следующее:

- С помощью метода *distans transform* ищется центр ладони.
- Считаются все векторы, идущие с центра ладони и до дефекта. Дальше они ортонормируются.
- Среди посчитанных векторов ищется средний.
- Те вектора, образующие тупой угол со средним отсекаются.

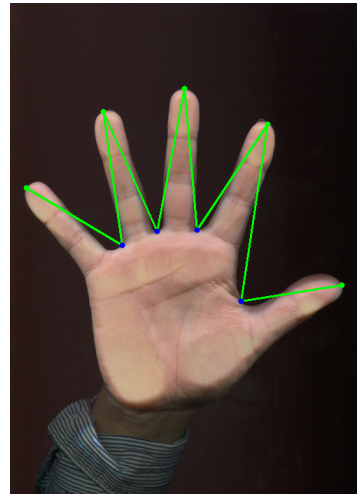
Результат на рис. 4.

2.3 Поиск большого пальца.

Были сохранены начало и конец дефекта. Конец одного дефекта - это начало другого. Усредняя их мы получаем точку на кончике пальца.



До фильтрации дефектов



После фильтрации дефектов.

Рис. 4: Дефекты до и после фильтрации.

Далее, для того, чтобы определить большой палец выполнялось следующее:

- Проверялось условие того, что конец одного дефекта - это начало другого. Если нарушено это условие, то мы на границе (либо мизинец, либо большой палец). Таким образом можно отсортировать дефекты так, чтобы первым был дефект, отвечающий либо за мизинец, либо за большой палец.
- Длина между основанием мизинца кончиком безымянного пальца меньше, чем расстояние между основанием большого пальца и указательного. Эта информация использовалась для того, чтобы найти большой палец.

Результат работы на рис. 4. Таким образом сформировался вектор признаков для ладоней. Дальше идет кластеризация.

3 Кластеризация и результаты.

Кластеризация проводилась методом К-средних. Для разного числа кластеров был нарисован график зависимости между числом кластеров и функционалом качества. Подробнее на рис. 5.

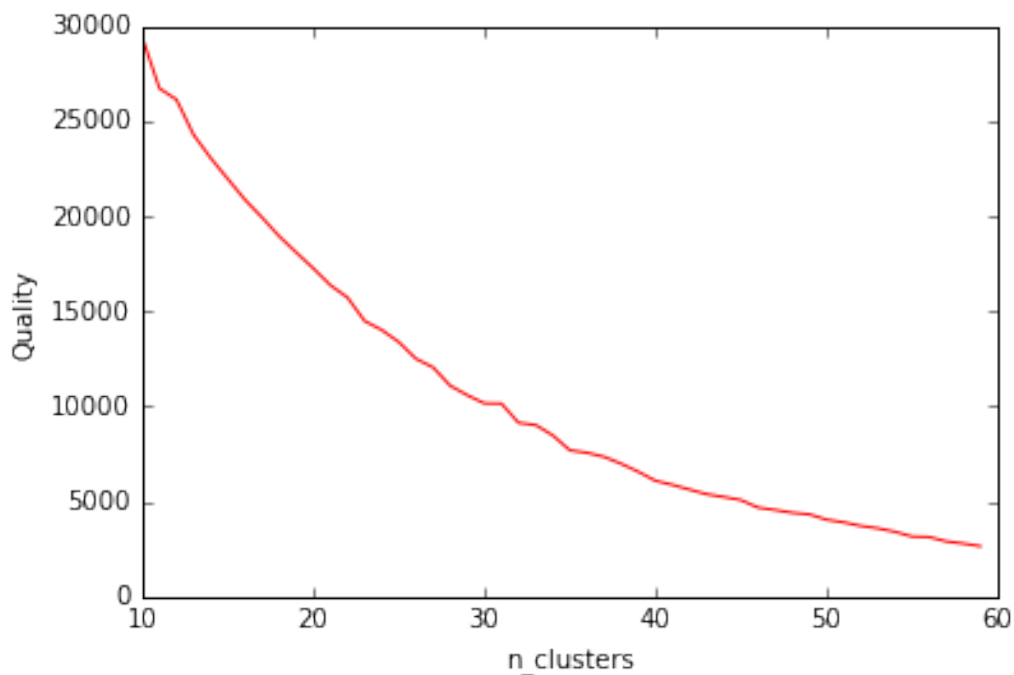


Рис. 5: График зависимости.

Как видно из графика, что чем больше количество кластеров, тем лучше качество, однако это сводится к переобучению. Поэтому было выбрано число кластеров равное 33. Результаты на Табл. 2. Результаты поиска близких ладоней на Табл. 1

Человек	Изображения ладоней
1	012 013 014 015 097
2	026 028 122
3	150 151 152
4	051 052 053 054
5	066 128 129
6	092 112 113 114
7	049
8	009 010 065 141
9	144 145 146
10	016 017 020 046 068
11	001 002 037 039
12	111
13	008 067 109
14	031 063 096
15	105 106 107 142
16	027 029
17	138
18	022 023 035 036
19	126 127
20	011 056 057 076 077 079 081 086
21	071
22	003 004 005 006 007 099 155 157
23	047 050 060
24	064
25	118 123
26	034 120 124
27	095
28	135
29	055 082 090
30	041
31	088 091
32	093
33	018 019 021 024 078

Таблица 2: Таблица кластеров.

Имя образца	Соседи	Имя образца	Соседи	Имя образца	Соседи
001	002 037 090	039	037 002 001	095	109 008 067
002	001 037 145	041	060 105 049	096	063 093 031
003	006 007 005	046	020 018 016	097	007 003 014
004	006 003 007	047	050 060 146	099	012 014 003
005	007 003 006	049	047 060 041	105	142 107 106
006	003 004 007	050	047 060 146	106	142 105 107
007	005 003 155	051	052 054 053	107	105 088 091
008	067 066 057	052	051 054 053	109	092 067 113
009	011 010 065	053	054 052 051	111	096 031 063
010	009 011 065	054	051 052 078	112	114 113 092
011	079 009 081	055	053 052 106	113	114 112 092
012	014 013 006	056	086 076 057	114	112 113 092
013	012 014 015	057	076 056 081	118	123 122 113
014	012 013 006	060	047 145 050	120	124 034 028
015	013 005 014	063	096 034 066	122	123 092 026
016	017 020 046	064	141 066 009	123	122 118 092
017	016 020 046	065	009 010 155	124	034 120 028
018	021 019 020	066	128 129 011	126	127 124 120
019	021 018 078	067	008 109 129	127	126 122 123
020	046 018 016	068	020 011 079	128	129 066 079
021	019 018 020	071	150 151 050	129	128 066 008
022	023 035 026	076	077 079 086	135	003 157 107
023	022 035 124	077	076 079 086	138	141 065 064
024	086 021 079	078	077 076 079	141	009 064 065
026	028 029 022	079	076 077 011	142	105 106 107
027	029 028 036	081	079 086 082	144	004 145 146
028	026 124 029	082	090 081 007	145	144 146 002
029	027 028 026	086	077 079 076	146	004 144 006
031	063 096 009	088	091 107 157	150	151 152 071
034	124 120 028	090	082 005 007	151	150 152 071
035	022 023 026	091	088 107 105	152	150 151 071
036	035 120 028	092	112 114 113	155	007 006 005
037	002 001 039	093	096 063 092	157	107 086 155

Таблица 1: Таблица ближайших соседей.