

Data oddania: _____

Ocena: _____

Hubert Marcinkowski 214942
Artur Wróblewski 214985

Zadanie 1

1. Cel

Zadanie polegało na stworzeniu szkieletu uniwersalnej aplikacji do rozpoznawania obiektów. W tym celu należało przygotować odpowiedni zestaw cech do klasyfikacji oraz opracować moduł jej dokonujący z wykorzystaniem zadanej metryki. W celu sprawdzenia stworzonej aplikacji należało wykorzystać ją do klasyfikacji obiektów 2 baz danych: MNIST oraz STaR.

2. Opis implementacji

Aplikacja została napisana w języku C++, jako środowisko zostało wykorzystane Qt. Pozwoliło to na ograniczenie ilości wymaganych klas, gdyż cały wygląd programu opracowany został za pomocą języka QML. Jedynie obliczenia wykonywanych metod wymagały kodu w języku C++, a co za tym idzie zaimplementowania przechowywujących je klasy. Z racji na tą uproszczoną strukturę uznano, iż budowa diagramu UML jest zbędna.

Program pozwala na operacje na obrazie w skali odcieni szarości. Mając obraz wejściowy użytkownik jest w stanie dokonywać przetwarzania obrazu wyjściowego i obserwować te zmiany zarówno na samym obrazie jak i jego histogramie. W momencie zaakceptowania zmian obraz wejściowy zamieniany jest z tym zmodyfikowanym, co pozwala na przeprowadzenie na nim dalszych operacji.

3. MNIST

Baza MNIST zawiera ręcznie pisane cyfry. Składa się ze zbioru uczącego (60 000 przykładów) oraz testowego (10 000). Każdy z przykładów to obraz pojedynczej cyfry.

3.1. Zestaw cech

Dla bazy MNIST zaproponowaliśmy użycie 8 cech:

Uwaga: Każdy obraz traktowaliśmy jako binarny tj. piksel był uznawany za element, gdy jego jasność była większa niż 10 (w skali 0-255). Wszystkie inne uznawane są za tło.

Uwaga: Przy pierwszych 6 poniższych cechach dla każdej cyfry wyznaczaliśmy bryłę brzegową (ang. bounding box). Dzięki temu wyeliminowaliśmy przesunięcia cyfr w każdym kierunku.

- **Ilość jasnych pikseli w dolnej połowie cyfry**
Cyfry takie jak 6 czy 9 mają różną ilość pikseli w górnej oraz dolnej połowie, dzięki czemu wraz z kolejną cechą można całkiem dobrze je odróżnić od np. 1, 8 czy 0.
- **Ilość jasnych pikseli w górnej połowie cyfry**
Analogicznie do poprzedniej cechy.
- **Ilość ciemnych pikseli od lewej krawędzi cyfry do lewej krawędzi obrazu**
Sprawdzając każdy z wierszy zliczaliśmy ciemne piksele, aż do napotkania pierwszego piksela cyfry (jasnego). Połączenie tej oraz 4 kolejnych cech pozwoliło na rozpoznanie kształtu cyfr z każdej strony. Metoda ta nie jest jednak odporna na obroty cyfr oraz ich zmienną wysokość lub szerokość. Opis zliczania jest analogiczny dla 3 kolejnych cech.
- **Ilość ciemnych pikseli od prawej krawędzi cyfry do prawej krawędzi obrazu**
- **Ilość ciemnych pikseli od górnej krawędzi cyfry do górnej krawędzi obrazu**
- **Ilość ciemnych pikseli od dolnej krawędzi cyfry do dolnej krawędzi obrazu**
- **Odległość euklidesowa pikseli od środka cyfry**
Środek wyznaczamy przy użyciu średniej arytmetycznej. Dzięki temu obliczyliśmy zwartość cyfry. Oczywiście część cyfr jest bardzo zbliżona pod tym względem.
- **Stosunek wysokości do szerokości**
Obliczyliśmy sumy odległości wszystkich pikseli od środków w dwóch kierunkach (od środka szerokości oraz środka wysokości). Ich stosunek pozwolił obliczyć "smukłość" cyfry.

3.2. Niewykorzystane cechy

Początkowo próbowaliśmy użyć jeszcze kilku innych cech - odrzuciliśmy je jednak z powodu braku znacznego polepszenia wyników (a czasem nawet pogorszenia) oraz wydłużania czasu obliczeń.

- **Ilość jasnych pikseli cyfry**
Statystycznie każda cyfra powinna mieć inną ilość pikseli składowych, niestety nie sprawdziło się to ze względu na różną wielkość cyfr.
- **Pole bryły otaczającej**
Celem było odróżnienie cyfr zajmujących mniejszą powierzchnię np. 1 oraz 0. Problemem był brak "odporności" na obroty cyfr.
- **Stosunek ilości jasnych pikseli do pola powierzchni bryły otaczającej**
- **Współczynniki kształtu oparte na momentach konturów**
Do wyznaczenia konturów użyliśmy filtracji liniowej.

3.3. Wyniki

Tabela 1. Wyniki jakości klasyfikacji oraz czasu obliczeń k-NN dla bazy MNIST dla różnych wartości k

k	jakość	czas[s]
1	73.67	69.783
3	76.08	67.983
5	77.90	68.175
7	78.01	68.076
9	78.31	67.515
11	78.58	68.414
13	78.26	67.970
15	77.94	67.725
19	78.00	67.416
35	77.54	67.814
99	75.70	68.476

Tabela 2. Wyniki jakości klasyfikacji k-NN dla bazy MNIST dla różnego zestawu cech

wybrane cechy	jakość
1	21.19
2	21.69
3	18.96
4	17.19
5	18.86
6	21.54
7	24.84
8	22.58
7,8	35.62
3,4	29.54
1,2,6,7,8	66.97
1,3,4,5,6	65.59
3,4,5,6,7	67.26
3,4,5,6,8	63.99

Tabela 3. Macierz pomyłek k-NN dla bazy MNIST dla wszystkich cech oraz $k = 11$

.	0	1	2	3	4	5	6	7	8	9	success ratio
0	925	0	15	3	11	1	3	0	18	4	94.38
1	0	1086	8	7	4	5	4	0	16	5	95.68
2	32	11	633	186	20	78	29	15	25	3	61.33
3	25	16	103	719	1	33	7	45	45	16	71.18
4	20	13	27	2	772	18	8	6	10	106	78.81
5	19	11	80	102	13	506	30	33	87	11	56.72
6	6	8	21	3	6	23	885	0	6	0	92.38
7	2	36	10	24	15	25	0	818	26	72	79.57
8	87	5	18	31	23	39	9	13	701	48	71.97
9	18	14	4	18	47	12	3	41	39	813	80.57

4. STaR

Jest to baza obrazów dziesięciu obiektów. Na zdjęciach występują 3 rodzaje tła, a pozycja, obrót oraz powiększenie obiektu są dobrane losowo.

4.1. Zestaw cech

Dla bazy STaR zdecydowaliśmy się skorzystać z tzw. momentów obiektu (obrazu), a dokładnie niezmienników przekształceń. Metoda ta pozwala na rozpoznawanie wzorów niezależnie od pozycji, rozmiaru czy obrotu. Jako cechy użyliśmy każdego z niezmienników - wzory podajemy poniżej:

$$\begin{aligned}
I_1 &= \eta_{20} + \eta_{02} \\
I_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
I_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
I_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
I_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
I_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
I_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned}$$

4.2. Niewykorzystane cechy

Początkowo planowaliśmy wykorzystać cechy stworzone dla bazy MNIST, jednak żadna z nich nie była "odporna" na przesunięcie, obrót lub skalowanie.

4.3. Wyniki

5. Wnioski