

Krzysztof Barden 210139 210139@edu.p.lodz.pl
Adam Troszczyński 210342 210342@edu.p.lodz.pl

Zadanie 1.: Sieci neuronowe cz.2

1. Cel

Część dalsza zadania nr 1. - Kompresja W tym zadaniu należy przeprowadzić proces kwantyzacji zbioru danych, przy czym dane te powinny reprezentować ramki obrazu poddawanego kompresji (ramki 4x4). Po kwantyzacji każdą ramkę obrazu należy zastąpić ramką odpowiadającego mu najbliższego wzorca i jako wynik zapisać tak przygotowany obraz. Proces kwantyzacji przeprowadzany jest za pomocą algorytmu K-srednich.

2. Wprowadzenie

Algorytm k-średnich to iteracyjny algorytm, w którym początkowo w przestrzeni z danymi rozmieszczone jest k centrów (centroidów). Następnie w każdej iteracji do każdego z centrów przyporządkowywane są dane wejściowe na podstawie ich odległości do centrum oraz aktualizowane jest położenie centrum, które wyznaczane jest jako średnia z przyporządkowanych do niego danych. Iteracje te powtarzane są tak długo aż położenie centrów ulegnie stabilizacji.

3. Opis implementacji

Do wykonania zadania został użyty język Python

Algorytm kompresji:

Algorytm wczytuje obraz i dzieli go na ramki 4x4. Następnie za pomocą algorytmu K-srednich zostaje wypełniony słownik zadaną ilością słów do których następnie przypisywane są podobne do słów ramki (najmniejsza odległość Euklidesowa).

4. Materiały i metody

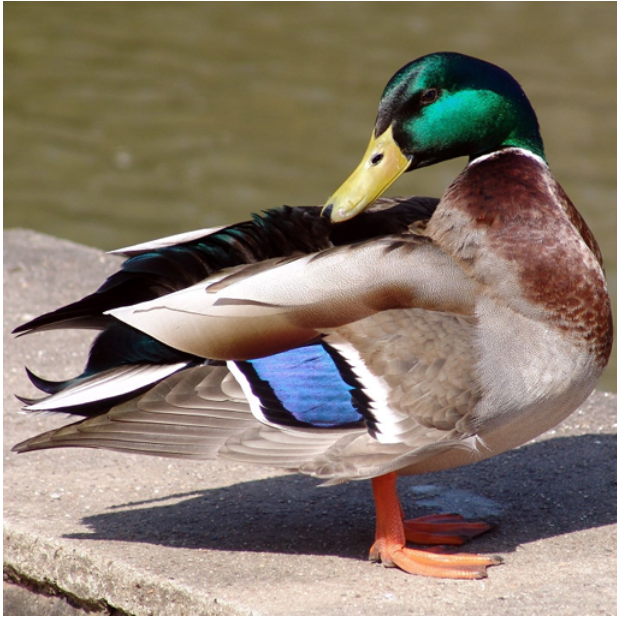
Obrazki Bazowe:
Lenna.png



chair.png



duck.png



house.png



iris.png



Eksperyment 3

Podpunkt 3.1

Obraz Lenna.png, 8 iteracji, 32 klastry(słowa), rozmiar ramki 4x4.

Obraz Lenna.png, 2 iteracje, 32 klastry(słowa), rozmiar ramki 4x4.

Obraz Lenna.png, 32 iteracje, 64 klastry(słowa), rozmiar ramki 8x8.

Podpunkt 3.2

Obraz chair.png, 4 iteracji, 64 klastry(słowa), rozmiar ramki 4x4.

Podpunkt 3.3

Obraz duck.png, 16 iteracji, 16 klastry(słowa), rozmiar ramki 4x4.

Obraz duck.png, 96 iteracji, 384 klastry(słowa), rozmiar ramki 4x4.

Podpunkt 3.4

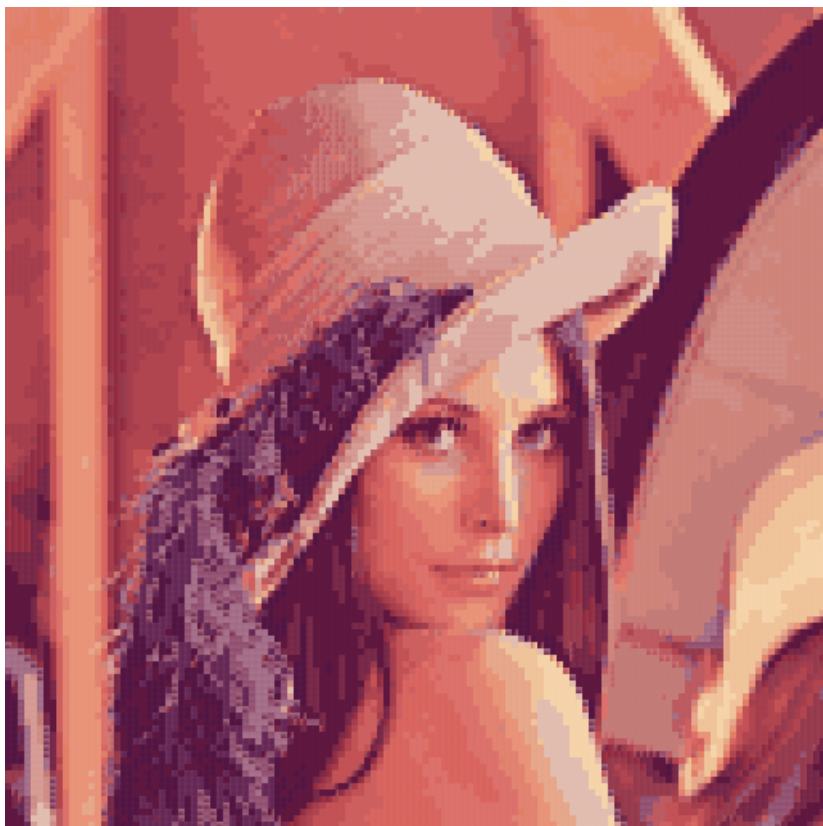
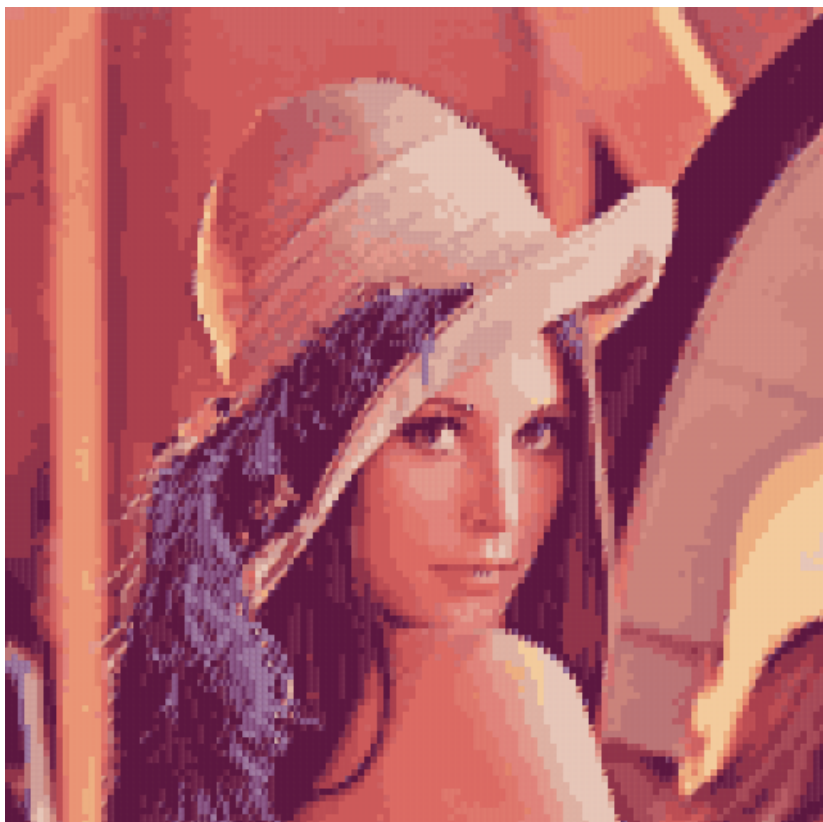
Obraz house.png, 8 iteracji, 32 klastry(słowa), rozmiar ramki 4x4.

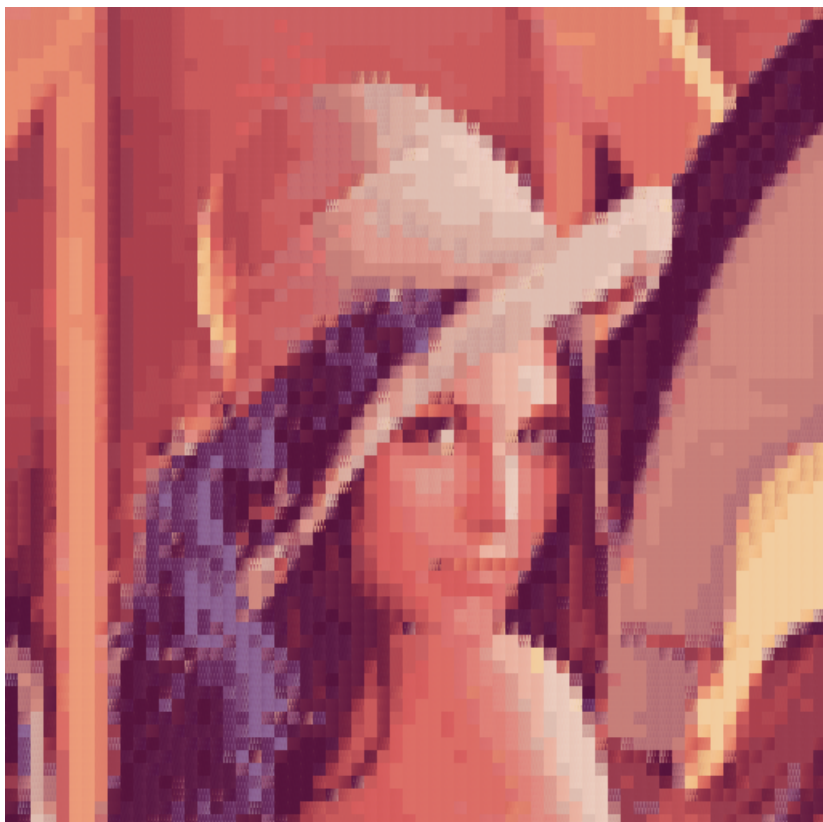
Podpunkt 3.5

Obraz Iris.png, 8 iteracji, 32 klastry(słowa), rozmiar ramki 4x4.

5. Wyniki

Podpunkt 3.1

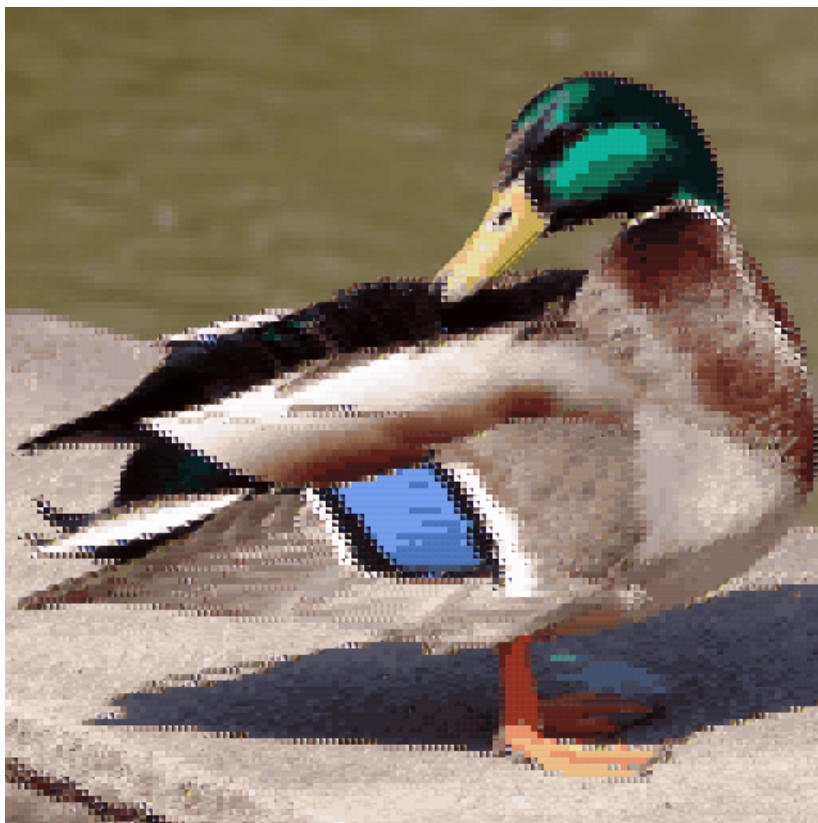




Podpunkt 3.2



Podpunkt 3.3



Podpunkt 3.4



Podpunkt 3.5



6. Dyskusja

Zwiększenie ilości klastrów pozwala na użycie większej ilości kolorów przy kompresji, ale też powoduje zwiększenie zajmowanej pamięci przez obrazek np: Obraz duck.png, 16 iteracji, 16 klastry(słowa). -> 41KB
Obraz duck.png, 96 iteracji, 384 klastry(słowa). ->142KB

Rozmiar ramki ma decydujący wpływ na stopień kompresji obrazu - im większy tym większa kompresja ale gorsza jakość. Powoduje też zmianę rozmiaru pliku. Im większy rozmiar tym mniej plik zajmuje pamięci.

Rozłożenie kolorów (słów w słowniku ramek) jest zależne od losowego rozmieszczenia centroidów w algorytmie k-srednich.

Ilość iteracji ma mały wpływ na wynik końcowy.

Obrazy o dużym kontraście (np kaczki na granicy białych i czarnych piór) nie są do końca poprawnie reprezentowane/przetwarzane.

7. Wnioski

- Losowość rozłożenia centroidów ma duże znaczenie na wynik kompresji, podobnie jak przy klasyfikacji danych.
- Na rozmiar zkompresowanego pliku wpływa ilość klastrów oraz rozmiar ramki.
- Ilość iteracji ma mały wpływ na wynik końcowy.
- Algorytm może mieć problemy przy przetwarzaniu obrazów o dużym kontraście.

Literatura

- [1] T. Oetiker, H. Partl, I. Hyna, E. Schlegl. *Nie za krótkie wprowadzenie do systemu L^AT_EX2e*, 2007, dostępny online.
- [2] <http://www.ai-junkie.com/ann/som/som1.html>.
- [3] <https://archive.ics.uci.edu/ml/datasets/Iris>
- [4] <https://archive.ics.uci.edu/ml/datasets/seeds>
- [5] <https://archive.ics.uci.edu/ml/datasets/Abalone>