SM LAB\_3 Sprawozdanie

Lewicki Maciej

1. Znaleźć 3 obrazy (Rozmiar minimalny 800x600 mogą być większe.) do przeprowadzanie analizy skuteczności kompresji, każdy ma reprezentować jedną z kategorii  (0.1 pkt):
   1. Rysunek techniczny,
   2. Skan dokumentu,
   3. Diagram, schematic

      Description automatically generatedKolorowe zdjęcie.

Table

Description automatically generated



1. Napisać kod dla dwóch rodzajów kompresji w formie koder i dekoder. Koder powinien zwracać pojedyncza zmienną, która zawiera również informację o oryginalnym rozmiarze kompresowanej informacji. Dekoder powinien przyjmować tą zmienną i zwracać oryginalną informację. Każdy kod, który będzie to realizował poza funkcją dekodującą lub przekazywał dodatkowe informacje, będzie oceniany negatywnie.
   1. Kompresja RLE. Dla dowolnego rodzaju danych. Można założyć że przekazujemy je w formie tablicy numpy. (0.3 pkt)

# Kompresja RLE:

Text

Description automatically generated

# Dekompresja RLE:

Text

Description automatically generated

1. Kompresja Quad Tree - Drzewo czwórkowe. Zakładamy, że na wejściu dostajemy obraz kolorowy lub w skali odcieni szarości. Kodujemy wszystkie warstwy na raz, więc kodowanie jest niezależnie od ich ilości, tylko sam kolor będzie miał tyle wartości ile jest warstw. (0.4 pkt)

# Text Description automatically generatedKompresja QuadTree:

# Text Description automatically generatedDekompresja QuadTree:

1. Krótkie sprawozdanie/raport z prac (0.1 pkt)
   1. Proszę poświęcić paragraf sprawozdania i opisać sposób wykorzystania pamięci przez wasze implementacje. Gdzie przechowywanych jest rozmiar obrazu w każdym z skompresowanych plików oraz jaką strukturą w pamięci jest drzewo.

# RLE:

Kompresja RLE odbywa się przy użyciu zwykłego ndarray, do którego wpierw wprowadzane są informację dotyczące rozmiaru zdjęcia a następnie wszystkie skompresowane dane.

# QuadTree:

QuadTree jest zrealizowane obiektowo na bazie „linked list”, w każdym „node” jest przechowywane: poziom, 4 wskaźniki na dzieci, rozdzielczość, średni kolor oraz to czy jest to liść. Po kompresji zachowujemy wszystkie poziomy dzięki czemu struktura pozwala nam obejrzeć obraz w różnych poziomach „ostrości” powoduje to jednak, że poziom kompresji dla wielu przykładów będzie poniżej 1, tj. kompresja będzie negatywna.

Oczywiście można napisać dodatkową metodę która zapisze wszystkie końcowe „node” lub wszystkie „node” na określonym poziomie w innej strukturze i odrzuci resztę, co poprawiłoby kompresję wielokrotnie. Przechowywanie całego drzewa ma sens jedynie gdy chcemy wykorzystać różne poziomy rozdzielczości.

* 1. Dla każdego z przygotowanych obrazów testowych przeprowadzić kompresję i dekompresję, każda z metod. Udowodnić poprawność (identyczność) danych po dekompresji z danymi źródłowymi. Sprawdzić skuteczność kompresji - zarówno jako jej stopień jak i jaki % oryginalnego obrazu stanowi obraz po kompresji.

Text

Description automatically generatedDowód poprawności danych zapewnia asercja przy wykonywaniu testów:

Text

Description automatically generatedObraz „schemat.png”

Text

Description automatically generatedObraz „scan.jpg”

Obraz „photo.jpg”

Text

Description automatically generated

# Demonstracja możliwości wyboru „poziomu” w QuadTree:

Treemap chart

Description automatically generated

A picture containing text, sunset, silhouette

Description automatically generatedA silhouette of a deer in front of a sunset

Description automatically generated with medium confidence