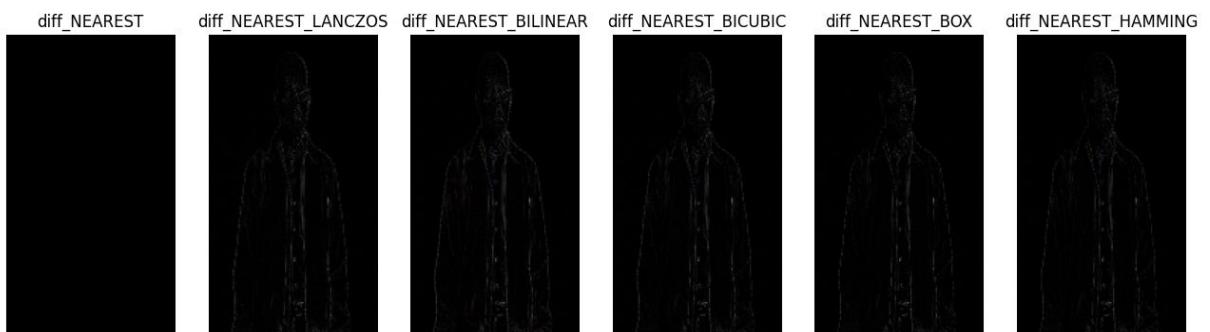


Krzysztof Krupicki – lab 10

Wybierz dowolny obraz przedstawiający postać (nazwa: `postac`)

1. Zastosuj metodę `resize` do obrazu `postac`. Utwórz 6 obrazów przyjmując skalę dla szerokości $s_w = 0.15$, skalę dla wysokości $s_h = 0.27$ oraz kolejno metody resamplingu 'NEAREST', 'LANCZOS', 'BILINEAR', 'BICUBIC', 'BOX', 'HAMMING'. Przedstaw na jednym diagramie plt (fig1.png) obrazy po przeskalowaniu i ich różnice w stosunku do NEAREST. Pobierz statystyki różnic i skomentuj.



Odpowiedź:

BOX / BILINEAR / HAMMING: Wykazują mniejsze różnice niż inne metody, ponieważ ich algorytmy są mniej agresywne w wygładzaniu krawędzi.

LANCZOS / BICUBIC: Generują największe różnice względem NEAREST. Wynika to z faktu, że algorytmy te stosują zaawansowaną interpolację, która wygładza krawędzie (antialiasing), podczas gdy NEAREST pozostawia ostre, schodkowane piksele.

Krzysztof Krupicki – lab 10

```
Statystyki dla różnicy obrazów NEAREST i NEAREST:  
tryb obrazu RGB  
rozmiar obrazu (120, 216)  
extrema [(0, 0), (0, 0), (0, 0)]  
count [25920, 25920, 25920]  
mean [0.0, 0.0, 0.0]  
median [0, 0, 0]  
stddev [0.0, 0.0, 0.0]  
Statystyki dla różnicy obrazów NEAREST i LANCZOS:  
tryb obrazu RGB  
rozmiar obrazu (120, 216)  
extrema [(0, 103), (0, 103), (0, 96)]  
count [25920, 25920, 25920]  
mean [2.9330246913580247, 2.7856095679012345, 2.8123456790123456]  
median [1, 1, 1]  
stddev [5.997773927534289, 5.750247016865465, 5.756609356896942]  
Statystyki dla różnicy obrazów NEAREST i BILINEAR:  
tryb obrazu RGB  
rozmiar obrazu (120, 216)  
extrema [(0, 109), (0, 107), (0, 100)]  
count [25920, 25920, 25920]  
mean [3.164699074074074, 2.95991512345679, 2.995679012345679]  
median [1, 1, 1]  
stddev [6.493235736218904, 6.1483378506820285, 6.1847255472250176]  
Statystyki dla różnicy obrazów NEAREST i BICUBIC:  
tryb obrazu RGB  
rozmiar obrazu (120, 216)  
extrema [(0, 104), (0, 103), (0, 96)]  
count [25920, 25920, 25920]  
mean [2.8368827160493826, 2.6909722222222223, 2.7254243827160494]  
median [1, 1, 1]  
stddev [5.979241160083945, 5.730134387967412, 5.741441716457597]  
Statystyki dla różnicy obrazów NEAREST i BOX:  
tryb obrazu RGB  
rozmiar obrazu (120, 216)  
extrema [(0, 102), (0, 101), (0, 98)]  
count [25920, 25920, 25920]  
mean [2.5982638888888889, 2.470833333333333, 2.517708333333333]  
median [1, 1, 1]  
stddev [5.777529781920638, 5.564284635340505, 5.56833901427468]  
Statystyki dla różnicy obrazów NEAREST i HAMMING:  
tryb obrazu RGB  
rozmiar obrazu (120, 216)  
extrema [(0, 96), (0, 95), (0, 89)]  
count [25920, 25920, 25920]  
mean [2.5659722222222223, 2.4227237654320986, 2.458719135802469]  
median [1, 1, 1]  
stddev [5.529470663096352, 5.30186468370438, 5.312589249342209]
```

Krzysztof Krupicki – lab 10

2. Porównaj dwa obrazy (statystyki różnicę) otrzymane w następujący sposób:
- Stosując metodę resize z wybraną metodą resamplingu powiększ fragment obrazu **postac** przedstawiający głowę dwukrotnie na szerokość i trzykrotnie na wysokość
 - Stosując metodę crop wytnij ten sam fragment obrazu **postac** i otrzymany obraz powiększ dwukrotnie z tą samą metodą resamplingu na szerokość i trzykrotnie na wysokość



Krzysztof Krupicki – lab 10

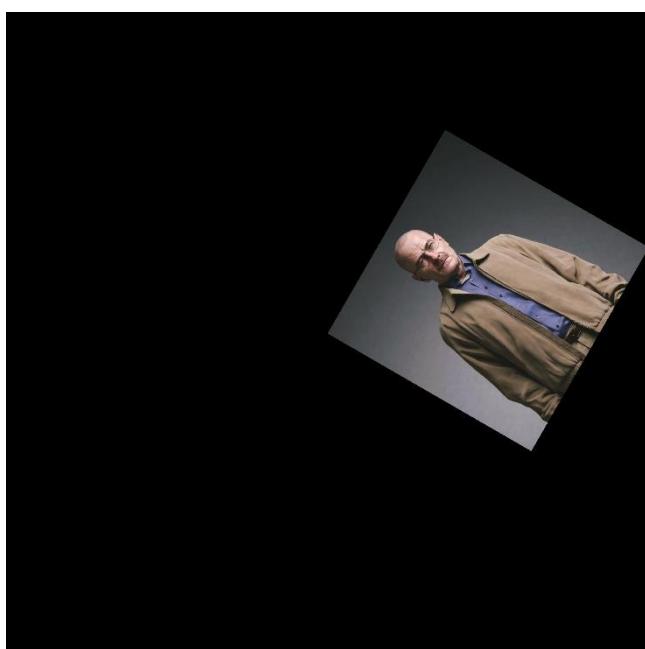
3. Obróć obraz

- a. o 60 stopni w lewo dobierając argumenty metody rotate tak, żeby widoczny był cały obraz **postac**, a nadmiarowy fragment był w kolorze czerwonym
- b. o 60 stopni w prawo dobierając argumenty metody rotate tak, żeby rozmiar obrazu **postac** się nie zmienił, a nadmiarowy fragment był w kolorze zielonym



4. Parametr expand nie działa dobrze, gdy center nie jest środkiem obrazu. Zaproponuj rozwiązanie, które pozwoli dokonać obrotu obrazu wokół punktu (0,0) tak, żeby widoczny był cały obraz. Wynik zapisz jako obraz **obrot.png**

Wskazówka: stworzyć nowy obraz o wymiarach takich, żeby punkt obrotu był jego środkiem.



Krzysztof Krupicki – lab 10

5. Czy przekształcenia `Image.TRANSPOSE` i `Image.TRANSVERSE` można otrzymać wykonując obroty i `Image.FLIP_LEFT_RIGHT`? Jeśli tak, napisz jak to zrobić.

Odpowiedź:

Oba przekształcenia można uzyskać za pomocą obrotu i odbicia lustrzanego:

`Image.TRANSPOSE` (odbicie względem głównej przekątnej, zamiana x na y): Wykonaj obrót o 270 stopni w lewo (lub 90 w prawo) i `Image.FLIP_LEFT_RIGHT`.

`Image.TRANSVERSE` (odbicie względem drugiej przekątnej): Wykonaj obrót o 90 stopni w lewo i `Image.FLIP_LEFT_RIGHT`.

6. Z obrazu `motylek.png` utwórz maskę o takich wymiarach, by na wybranym obrazie `postac` estetycznie wkleić trzy motylki (w normalnym położeniu, obrócony o 60 stopni w prawo, obrócony o 60 stopni w lewo). Skorzystaj z wklejania z maską obrazu `kolorowe_tlo.png` w obraz `postac`. Uwaga: można korzystać z `resize`, `rotate` i `convert`.

