

SIGK - Projekt 1

Modyfikacja obrazów

Autor: Łukasz Dąbała

1 Wymagania projektu

W ramach projektu należy stworzyć program, który będzie realizował opisane w temacie funkcje. Projekt jest zadaniem zespołowym, gdzie każdy zespół składa się z 2 osób.

Głównym językiem programowania powinien być język Python wraz z frameworkiem przeznaczonym do sieci neuronowych: Pytorch/Tensorflow.

Za projekt można uzyskać maksymalnie $x \times 10p.$, gdzie x to liczba osób w zespole. Każdy z członków zespołu może dostać maksymalnie 10 punktów.

Ocenie w ramach projektu podlegają:

1. Działanie programu - realizacja funkcji (7 p.)
2. Dokumentacja dokonanych eksperymentów oraz wizualizacja wyników (3 p.)

Projekt uznaje się za oddany w momencie prezentacji go prowadzącemu.

2 Modyfikacja obrazów

W ramach projektu należy zaproponować architekturę oraz wykorzystać ją do realizacji 2 z 4 zadań opisanych poniżej. W celu badania jakości stworzonego rozwiązania należy zastosować wszystkie metryki opisane w ramach wykładów tzn. SNE, PSNR, SSIM, LPIPS. Należy dążyć do rozwiązania lepszego niż bazowe zaproponowane w zadaniach.

Do każdego z zadań korzystamy ze zbioru danych DIV2K: <https://data.vision.ee.ethz.ch/cvl/DIV2K/>.

Zadanie oznaczone * jest uznawane za trudniejsze od pozostałych. Wykonanie go umożliwia późniejszą zmianę ścieżki projektowej na ścieżkę badawczą.

W ramach trenowania i walidacji wyników eksperymentów, korzystamy z obrazów w rozdzielczości 256×256 .

2.1 Zwiększanie rozdzielczości

Parametry danych wejściowych: obrazy w rozdzielczościach 32×32 , 64×64 .

Przygotowanie zbioru danych: należy napisać metodę do zmiany rozdzielczości każdego ze zdjęć np. korzystając z metody *resize* z biblioteki OpenCV. Należy udokumentować, jaki rodzaj interpolacji został użyty.

Porównanie wyników: metodą bazową z którą należy porównać stworzone rozwiązanie jest *resize* z OpenCV z interpolacją bikubiczną (ang. *bicubic interpolation*).

2.2 Odszumianie

Parametry danych wejściowych: współczynniki dla szumu gaussowskiego σ : 0.01, 0.03 dla znormalizowanych obrazków o wartościach w przedziale $< 0.0, 1.0 >$.

Przygotowanie zbioru danych: należy napisać metodę do zaszumiania obrazów korzystając np. z metody *random_noise* z biblioteki skimage.

Porównanie wyników: metodą bazową z którą należy porównać stworzone rozwiązanie jest *denoise_bilateral* z skimage.

2.3 Deblurowanie

Parametry danych wejściowych: wielkość kernela dla rozmycia gaussowskiego: 3×3 , 5×5 .

Przygotowanie zbioru danych: należy napisać metodę do rozmywania gaussowskiego zdjęć np. korzystając z metody *GaussianBlur* z biblioteki OpenCV.

Porównanie wyników: metodą bazową z którą należy porównać stworzone rozwiązanie jest *richardson_lucy* z skimage.

2.4 Inpainting*

Parametry danych wejściowych: losowe wycinanie obszarów o wielkościach: 3×3 , 32×32 .

Przygotowanie zbioru danych: należy napisać metodę do losowego wycinania fragmentów obrazów.

Porównanie wyników: metodą bazową z którą należy porównać stworzone rozwiązanie jest *inpaint* z OpenCV z parametrem *INPAINT_TELEA*.