31.10.2023

Sygnały i Obrazy Cyfrowe

Sprawozdanie 2 - Interpolacja Funkcji i skalowanie obrazów

Mateusz Marko, grupa nr 1, TN wt 17:05
Nr 273168, ISA

LINK Z KODEM DO OBYDWU ZADAŃ : https://github.com/MrMatier/Sygnaly-i-obrazy-cyfrowe/tree/LAB2

1. Cel ćwiczenia #1- Interpolacja Funkcji

Celem tego ćwiczenia jest wprowadzenie w podstawy interpolacji funkcji, w tym także zastosowanie tego procesu w kontekście cyfrowego przetwarzania obrazów. Interpolacja jest kluczowym narzędziem w tej dziedzinie, gdzie konieczne jest uzyskanie dodatkowych informacji (punktów) pomiędzy znanymi danymi. Sprawdzamy jak dodanie kolejnych punktów, ich rozmieszczenie, ilość i odległości między nimi wpływają na jakość interpolacji i wynik końcowy.

2. Wstęp teoretyczny

➢ Interpolacja

Interpolacja to proces estymacji wartości między znanymi danymi. W ogólnym kontekście przetwarzania sygnałów, może to oznaczać wyznaczanie wartości sygnału dla punktów, które leżą pomiędzy istniejącymi danymi. W kontekście przetwarzania obrazów, interpolacja jest używana do estymacji wartości pikseli obrazu na podstawie dostępnych informacji. Jest to szczególnie istotne przy zmianie rozdzielczości obrazów, czy też w innych przypadkach, gdy konieczne jest uzyskanie wartości pikseli dla pozycji, które nie są dokładnie pokrywające się z oryginalnymi danymi. Zatem interpolacja 2D odnosi się do procesu estymacji wartości w dwóch wymiarach, co jest istotne dla obrazów, które są reprezentowane jako dwuwymiarowe tablice pikseli. W przypadku obrazów cyfrowych, interpolacja 2D pozwala uzyskać wartości pikseli dla współrzędnych znajdujących się pomiędzy istniejącymi pikselami w obu kierunkach (poziomo i pionowo).

➢ Interpolacja funkcji za pomocą jednowymiarowej konwolucji

Interpolacja funkcji może być realizowana za pomocą jednowymiarowej konwolucji. Konwolucja jest operacją matematyczną, która łączy dwie funkcje, generując trzecią. W kontekście interpolacji jednowymiarowej, oblicza się nowe wartości funkcji za pomocą konwolucji z jądrem interpolacyjnym.

➢ Funkcje jądrowe

Funkcje jądrowe są używane w procesie konwolucji. Są to filtry, które decydują o tym, jak obliczane są nowe wartości na podstawie istniejących. Przykłady to proste jądra uśredniające lub bardziej zaawansowane jądra, takie jak jądra Gaussa.

➢ Kryterium MSE (Mean Squared Error)

Kryterium MSE to miara jakości interpolacji, która porównuje różnice między oryginalną funkcją/obrazem a funkcją/obrazem interpolowanym. Im niższą wartość MSE wskazuje tym lepsza jakość interpolacji.

$$MSE(y, \hat{y}) = \frac{1}{N}\sum_{i=0}^{N-1}(y_i - \hat{y}_i)^2,$$

Gdzie N to liczba próbek, y to funkcja oryginalna, a $\hat{y}$ to funkcja po dokonaniu interpolacji.

3. Przebieg ćwiczenia

W zadaniu trzeba było wygenerować dodatkowe punkty dla zadanej funkcji np. $sin(x)$, tak aby nowe punkty były możliwie blisko oryginalnej funkcji. W tym celu, rozpoczynając od zadanej liczby punktów np $N = 100$ na określonym odcinku (dla funkcji sinus, np na $[-\pi, \pi]$), należało stworzyć funkcję interpolowaną, posiadającą 2, 4 oraz 10 razy więcej punktów niż oryginalna funkcja. Następnie należało wykonać interpolację trzech funkcji za pomocą jedno-wymiarowej konwolucji, korzystając z wybranych jąder konwolucji h1, h2, h3, bądź h4. W kolejnym kroku trzeba było sprawdzić jakość zastosowanej konwolucji kryterium MSE oraz zbadać wpływ liczby oraz rozmieszenia punktów na jakość konwolucji. Można było także dodatkowo sprawdzić wpływ rozmieszczenia punktów na jakość interpolacji oraz porównać interpolację o dużej liczbie punktów z sekwencją interpolacji tworzących mniej punktów.

Przykładowe funkcje do interpolacji:

$$f_1(x) = \sin(x), \quad f_2(x) = \sin\,(x^{-1}), \quad f_3(x) = sgn(\sin(8x))$$

Podane funkcje jądrowe:

$$h_1(x) = \begin{cases} 1 & t \in [0,1) \\ 0 & t \notin [0,1) \end{cases}, \qquad h_2(x) = \begin{cases} 1 & t \in [-\frac{1}{2}, \frac{1}{2}) \\ 0 & t \notin [-\frac{1}{2}, \frac{1}{2}) \end{cases}$$

$$h_3(x) = \begin{cases} 1 - |t| \in [-1,1) \\ \;\;0 \quad t \notin [-1,1) \end{cases}, \qquad h_4(x) = \frac{\sin(x)}{x}$$

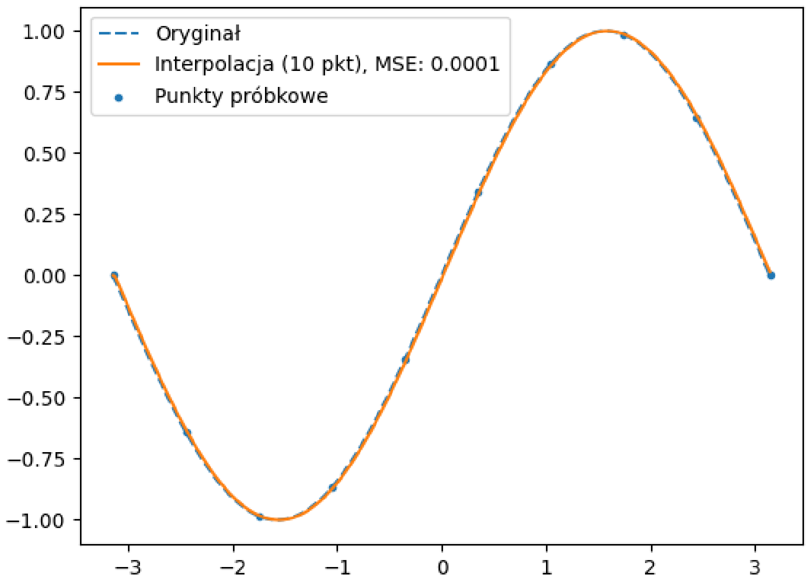
$$* - należy\ uważać\ na\ przypadki\ \frac{1}{x}, w\ których\ x = 0$$

W zadaniu chciałem porównać ze sobą dwie różne interpolacje ich jakość oraz dokładność również w zależności od wybrania jądra, dlatego poza wymaganą interpolacją przy użyciu konwolucji z różnymi jądrami, utworzyłem też drugą interpolacje tym razem średnią.

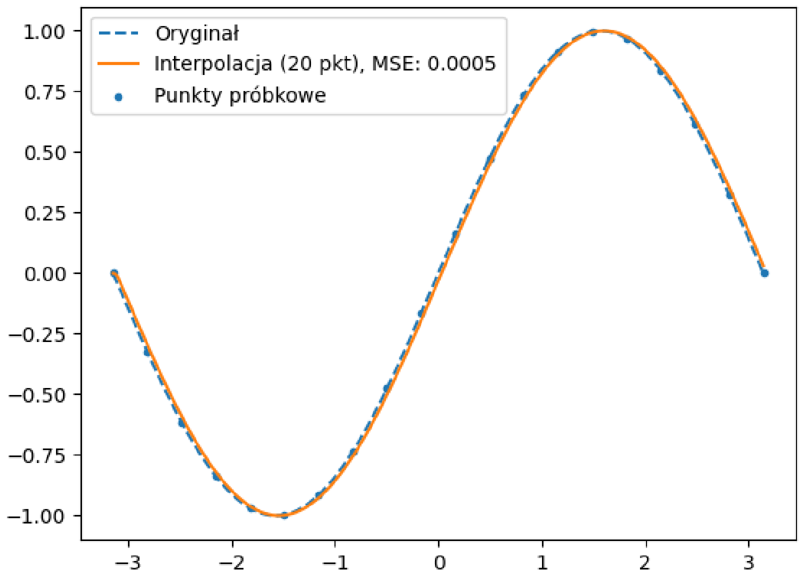
I. Interpolacja z jednowymiarową konwolucją

a) $\boldsymbol{f_1(x) = \sin(x)}$

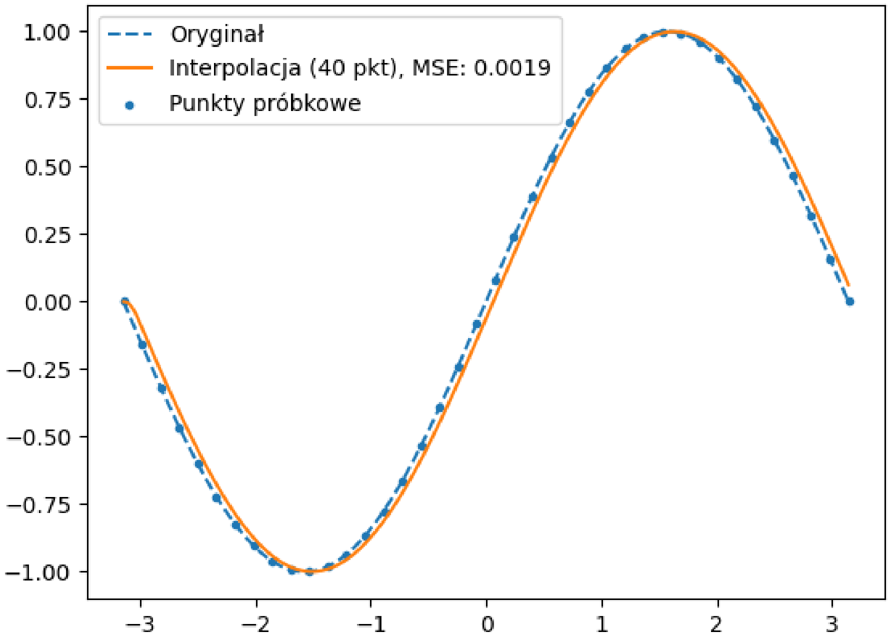
➢ Jądro $h_1(x)$



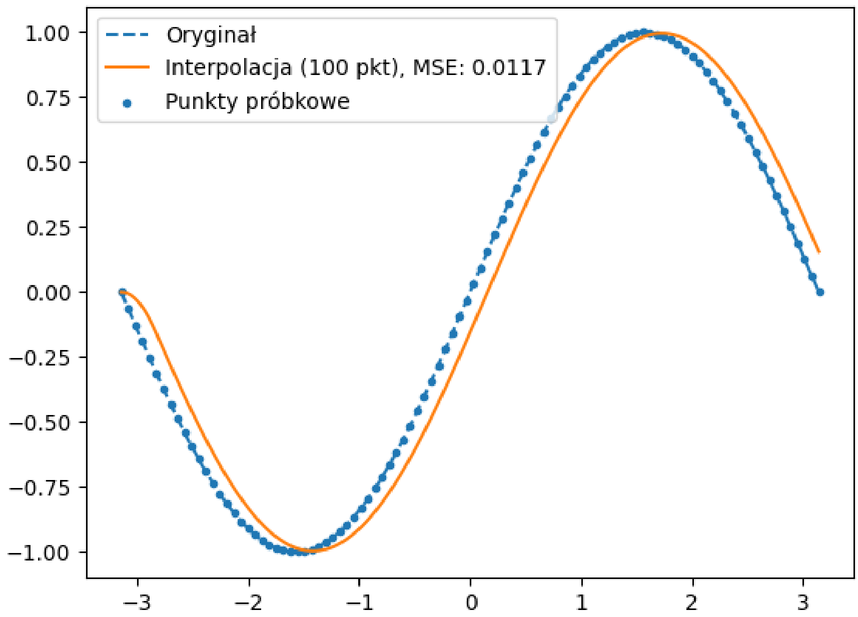
Rysunek 1 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 10 punktów próbkowania, kryterium MSE=0.0001



Rysunek 2 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 20 punktów próbkowania, kryterium MSE=0.0005

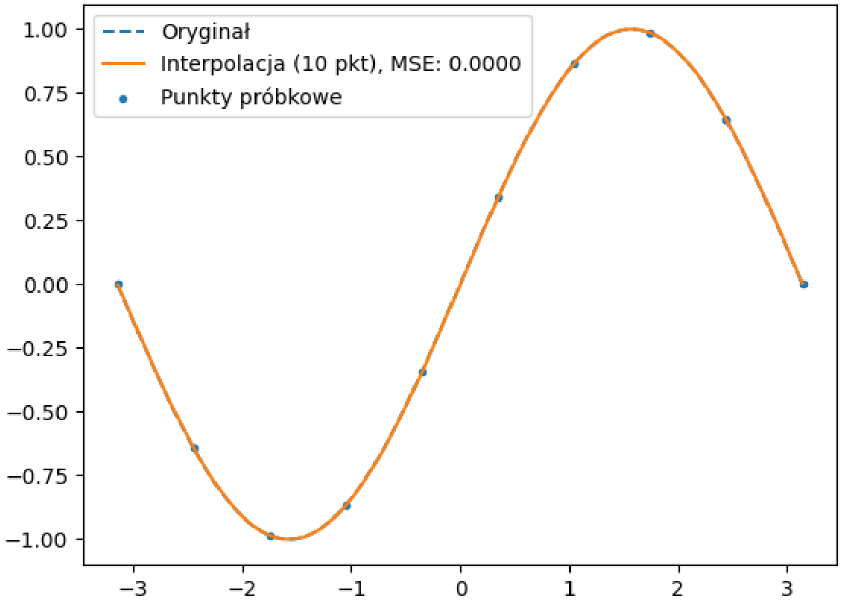


Rysunek 3 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 40 punktów próbkowania, kryterium MSE=0.0019

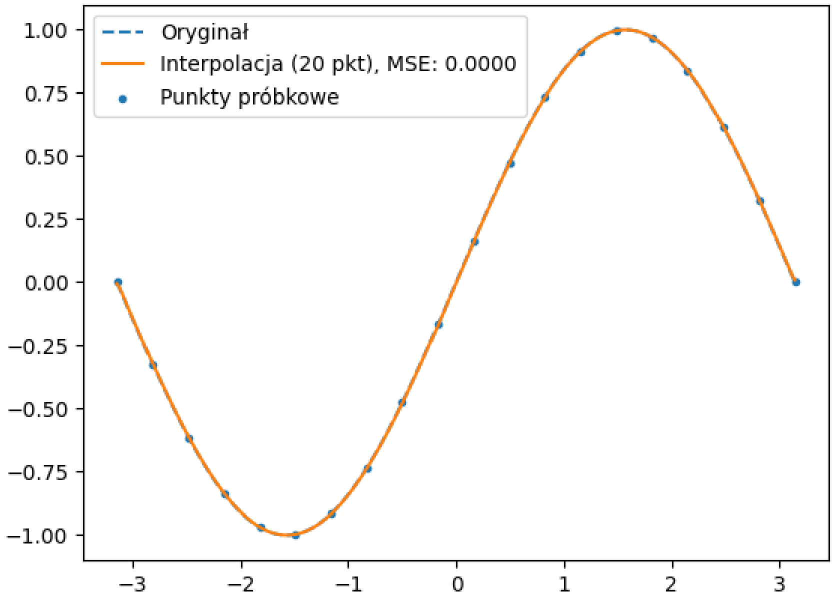


Rysunek 4 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 100 punktów próbkowania, kryterium MSE=0.0117

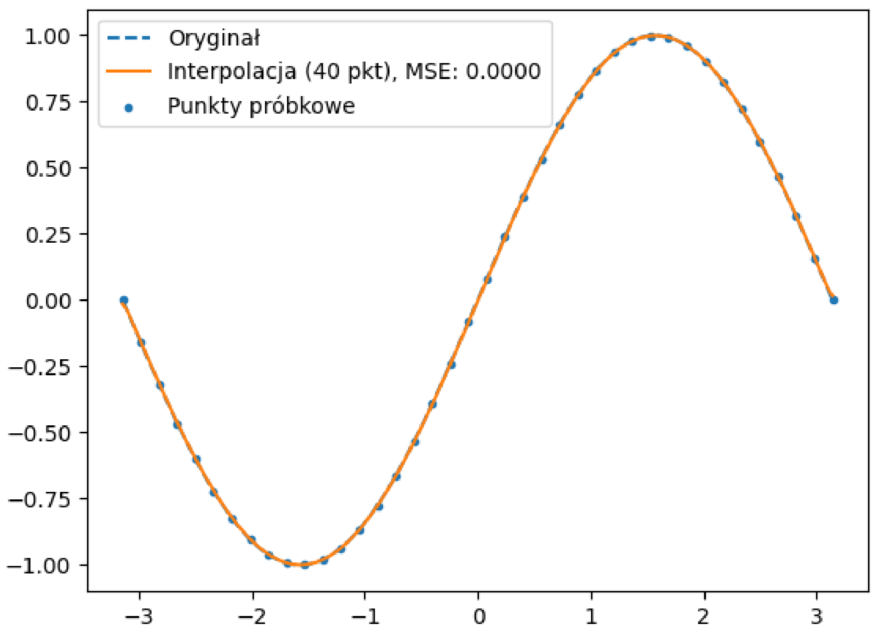
- Jądro $h_2(x)$



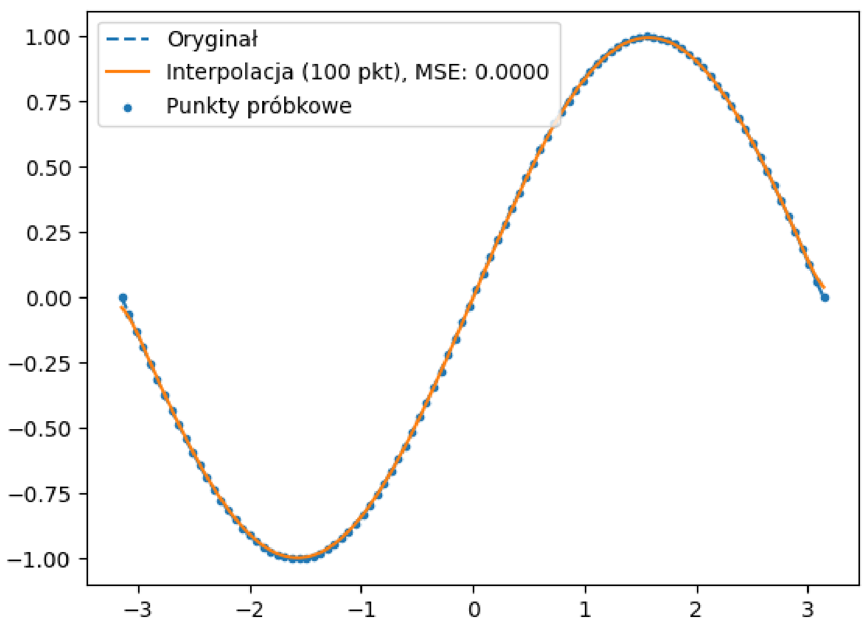
Rysunek 5 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 10 punktów próbkowania,
kryterium MSE<0.0001



Rysunek 6 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 20 punktów próbkowania,
kryterium MSE<0.0001

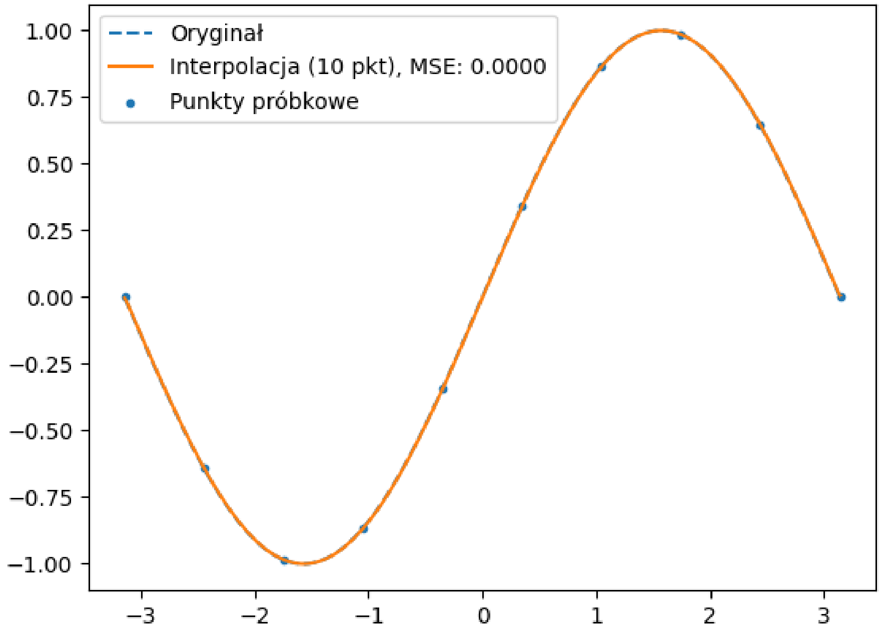


Rysunek 7 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 40 punktów próbkowania,
kryterium MSE<0.0001

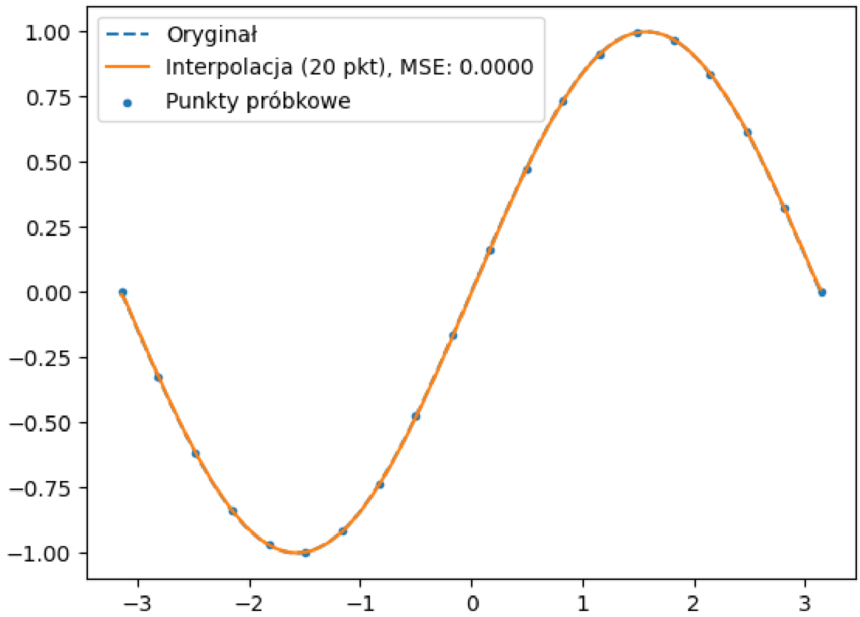


Rysunek 8 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 100 punktów próbkowania,
kryterium MSE<0.0001

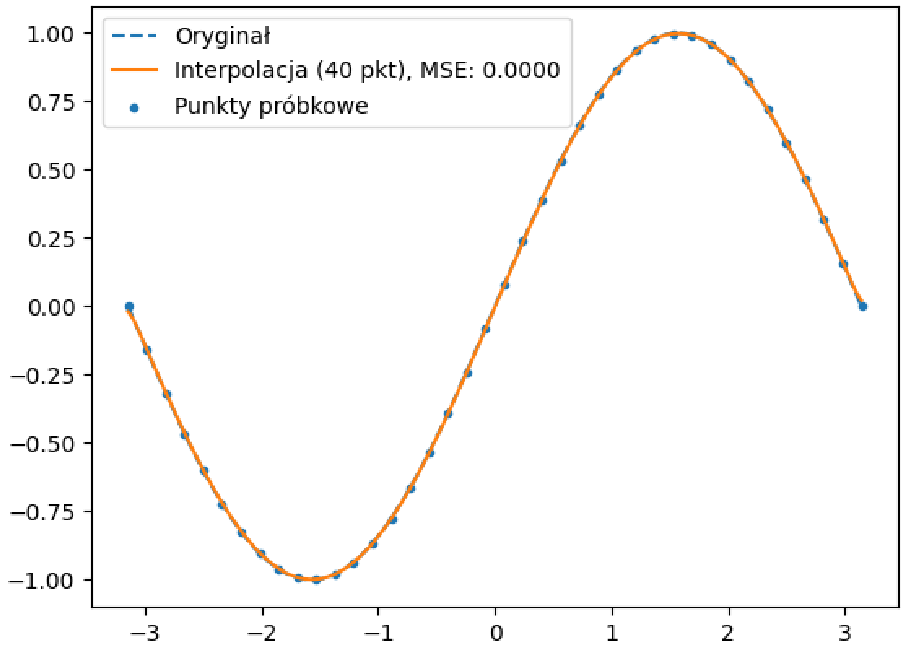
➢ Jądro $h_3(x)$



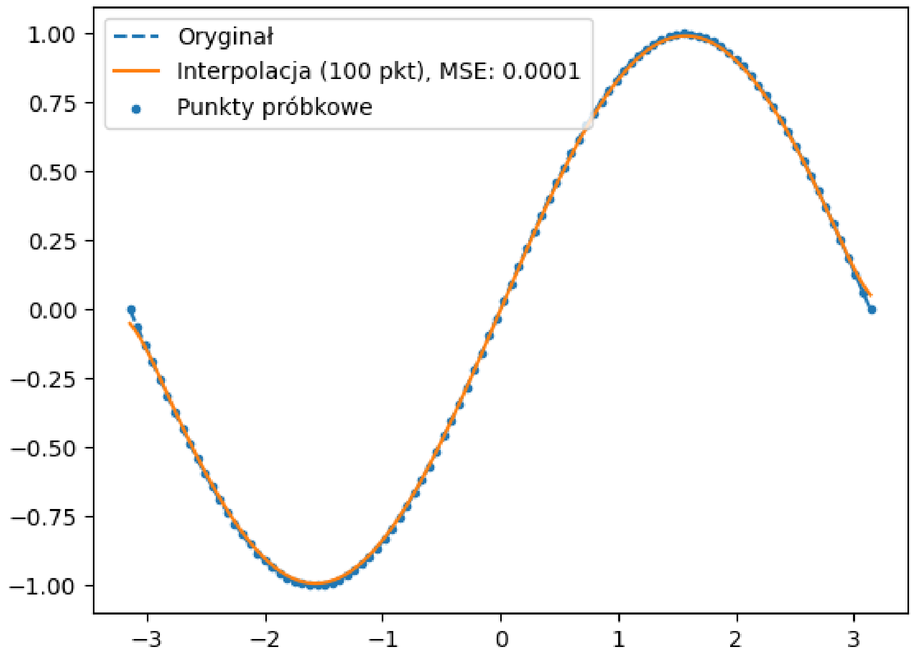
Rysunek 9 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 10 punktów próbkowania, kryterium MSE<0.0001



Rysunek 10 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 20 punktów próbkowania, kryterium MSE<0.0001



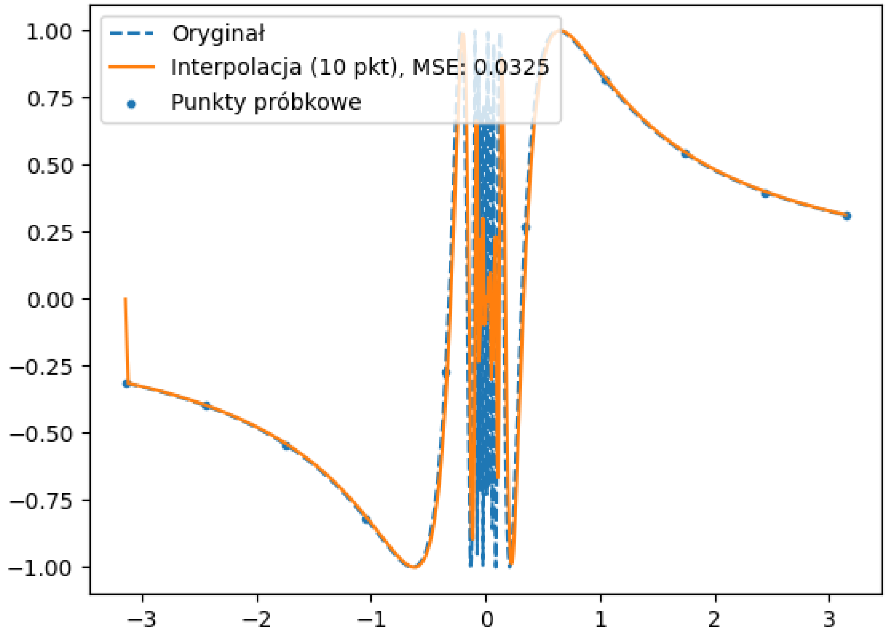
Rysunek 11 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 40 punktów próbkowania,
kryterium MSE<0.0001



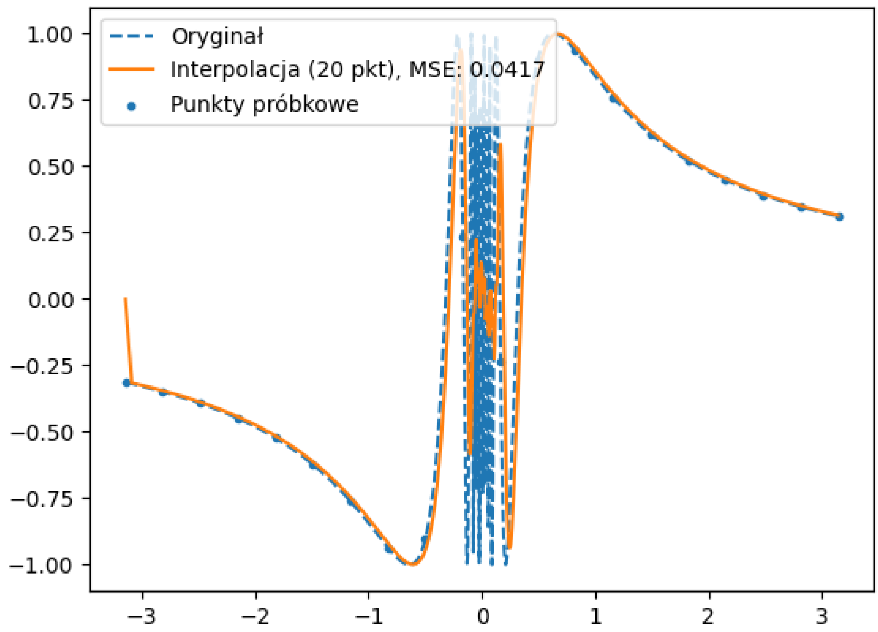
Rysunek 12 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 100 punktów próbkowania,
kryterium MSE=0.0001

b) $\boldsymbol{f_2(x) = \sin(x^{-1})}$

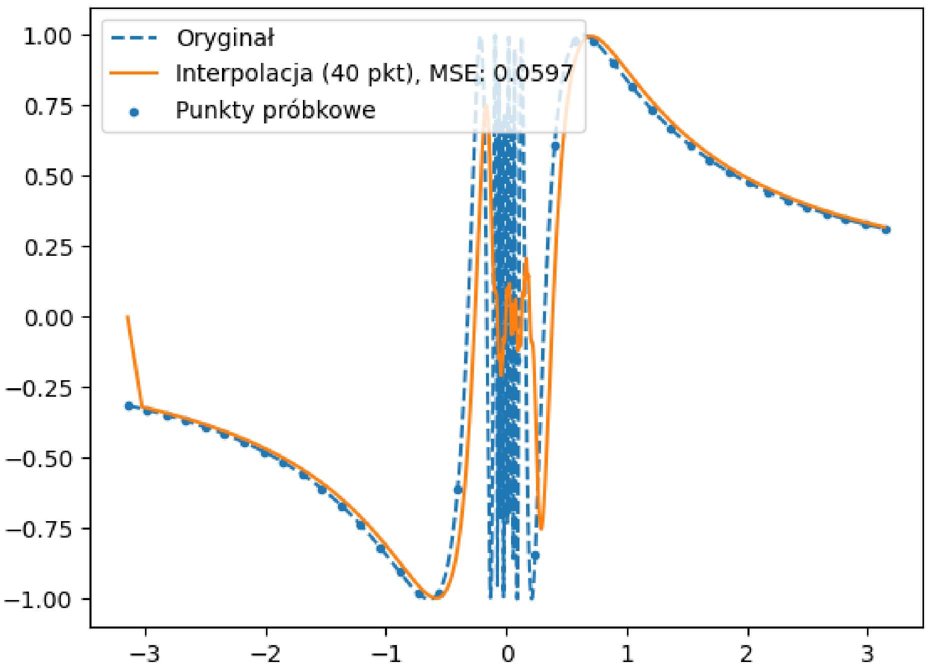
- Jądro $h_1(x)$



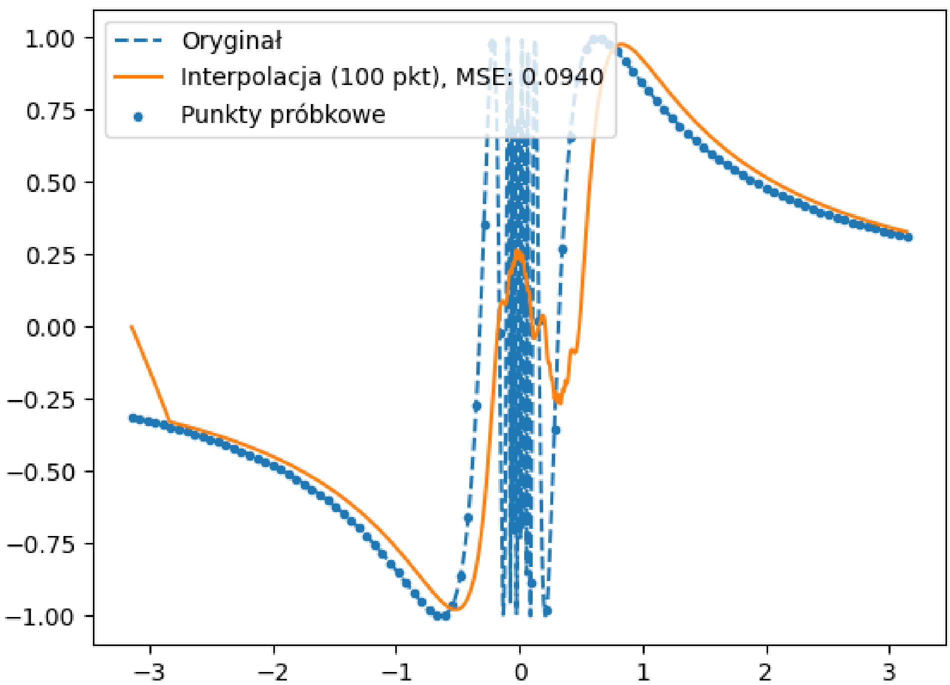
Rysunek 13 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 10 punktów próbkowania, kryterium MSE=0.0325



Rysunek 14 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 20 punktów próbkowania, kryterium MSE=0.0417

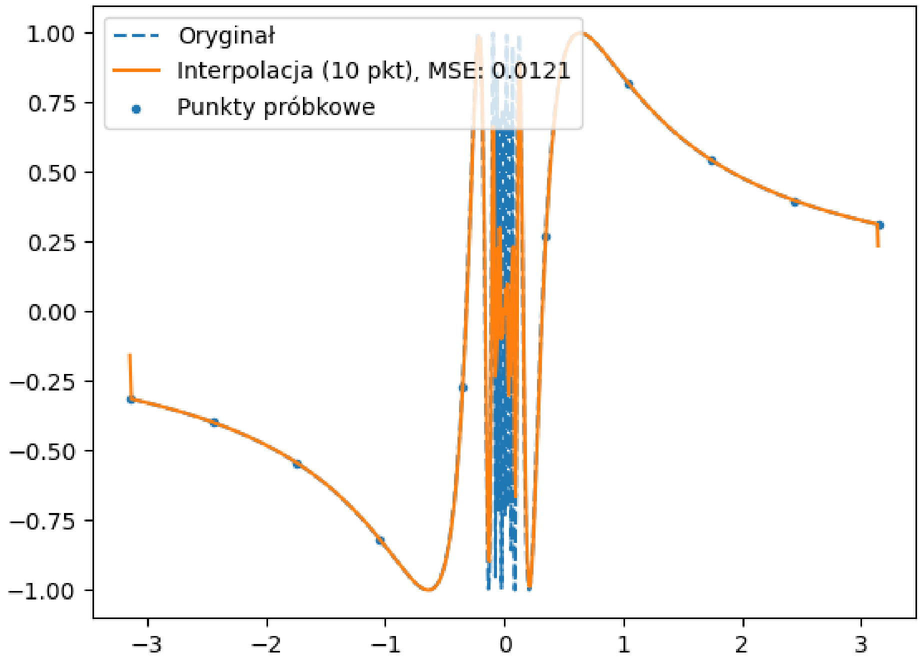


Rysunek 15 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 40 punktów próbkowania, kryterium MSE=0.0597

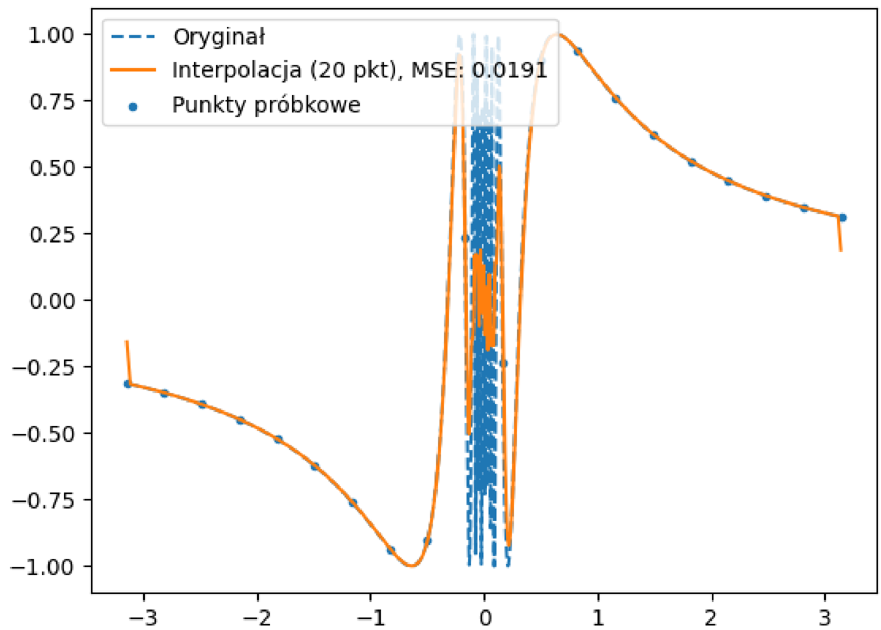


Rysunek 16 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 100 punktów próbkowania, kryterium MSE=0.0940

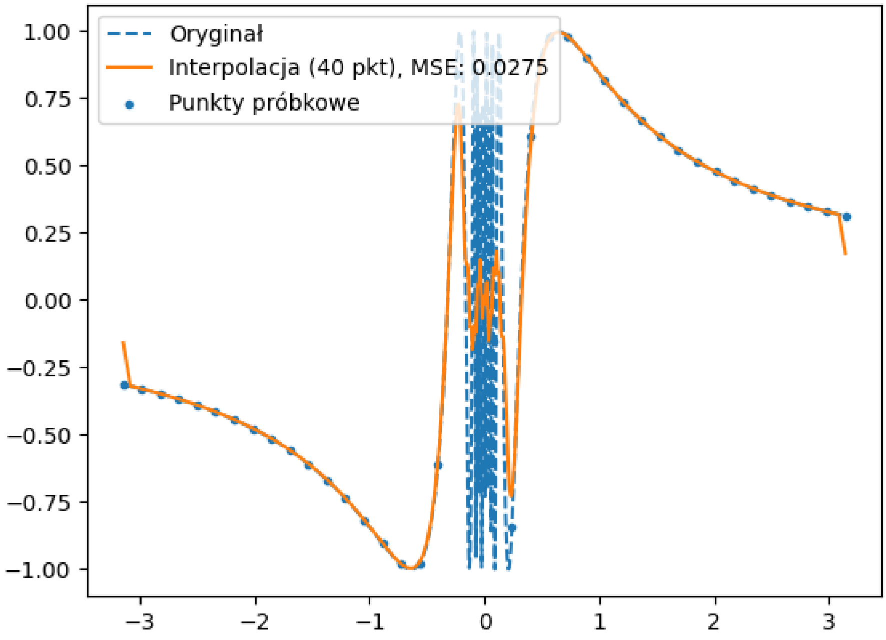
➢ Jądro $h_2(x)$



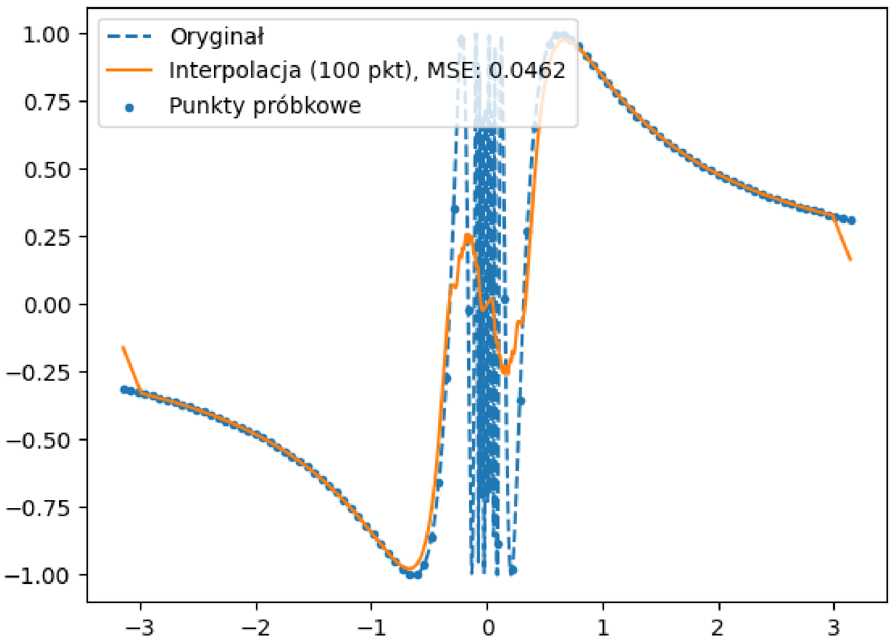
Rysunek 17 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 10 punktów próbkowania, kryterium MSE=0.0121



Rysunek 18 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 20 punktów próbkowania, kryterium MSE=0.0191

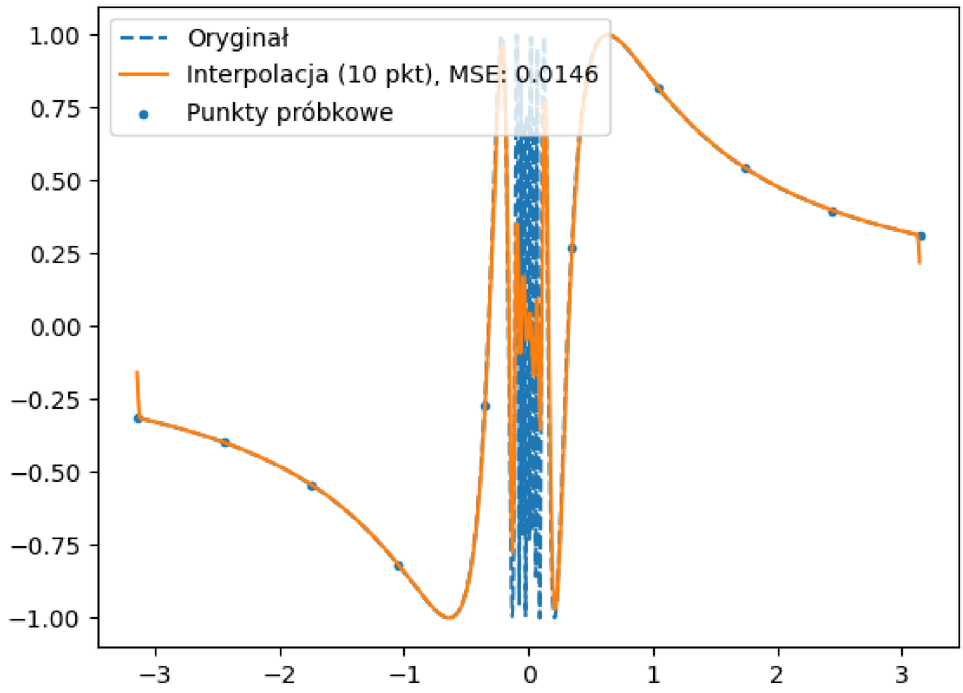


Rysunek 19 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 40 punktów próbkowania, kryterium MSE=0.0275

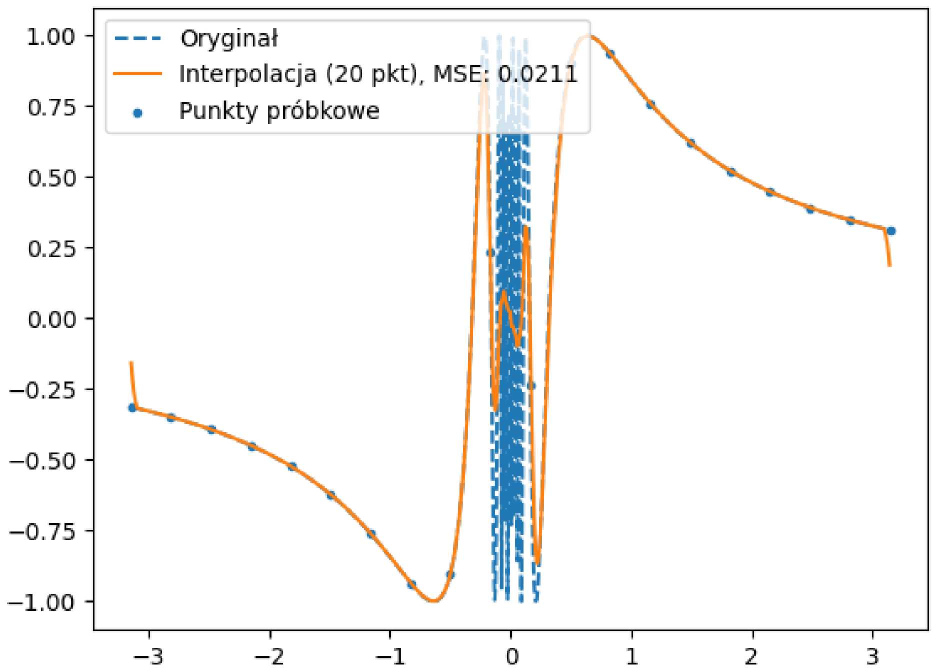


Rysunek 20 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 100 punktów próbkowania, kryterium MSE=0.0462

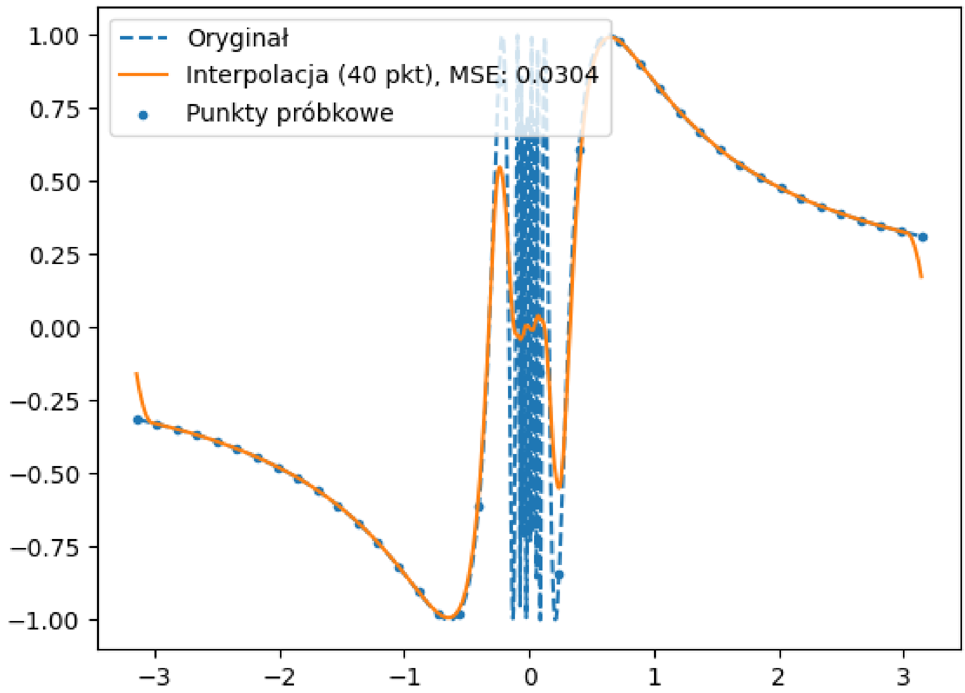
➢ Jądro $h_3(x)$



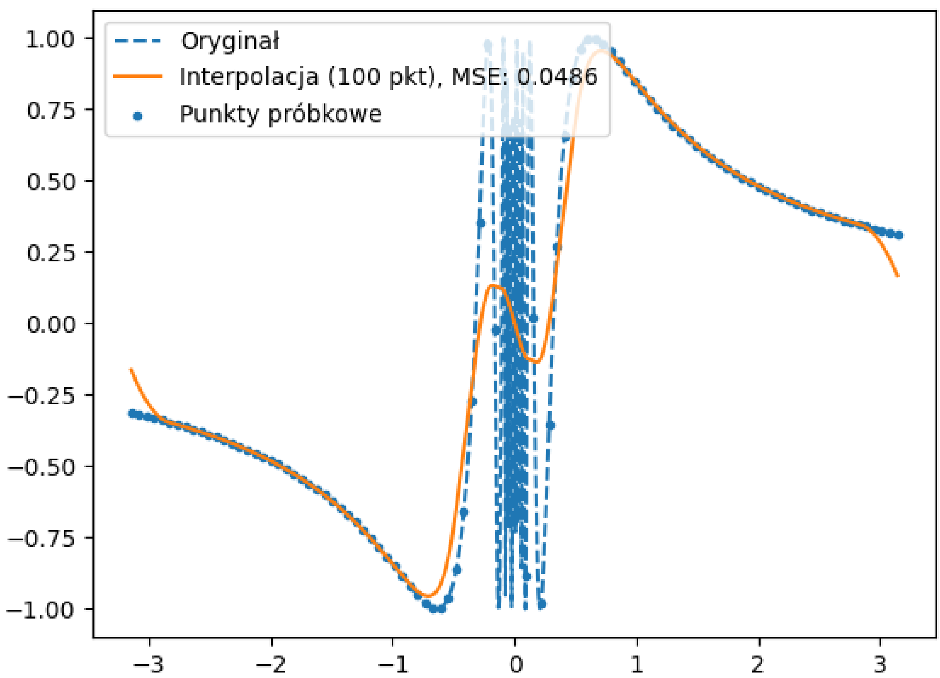
Rysunek 21 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 10 punktów próbkowania, kryterium MSE=0.0146



Rysunek 22 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 20 punktów próbkowania, kryterium MSE=0.0211



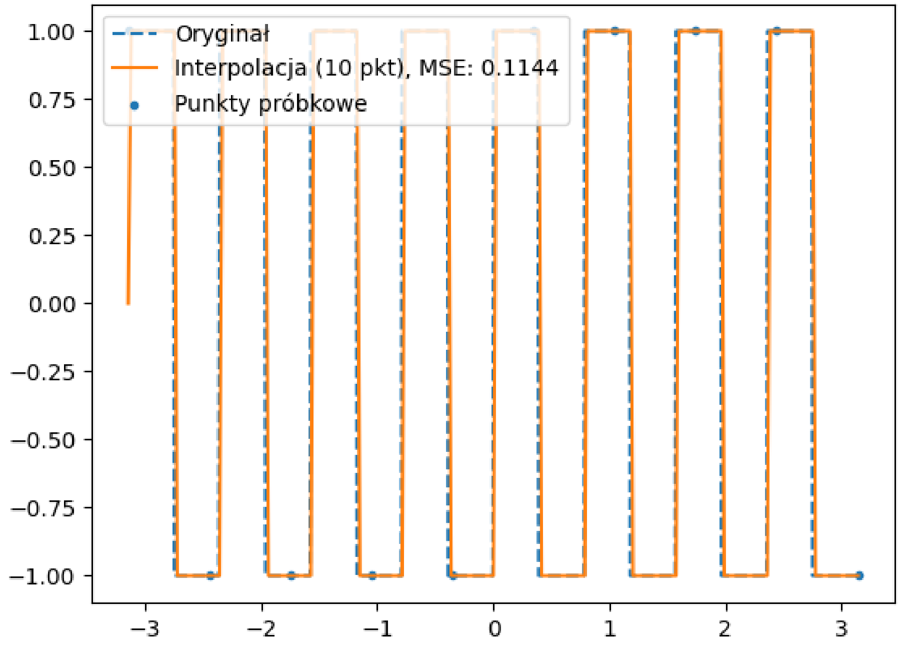
Rysunek 23 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 40 punktów próbkowania, kryterium MSE=0.0304



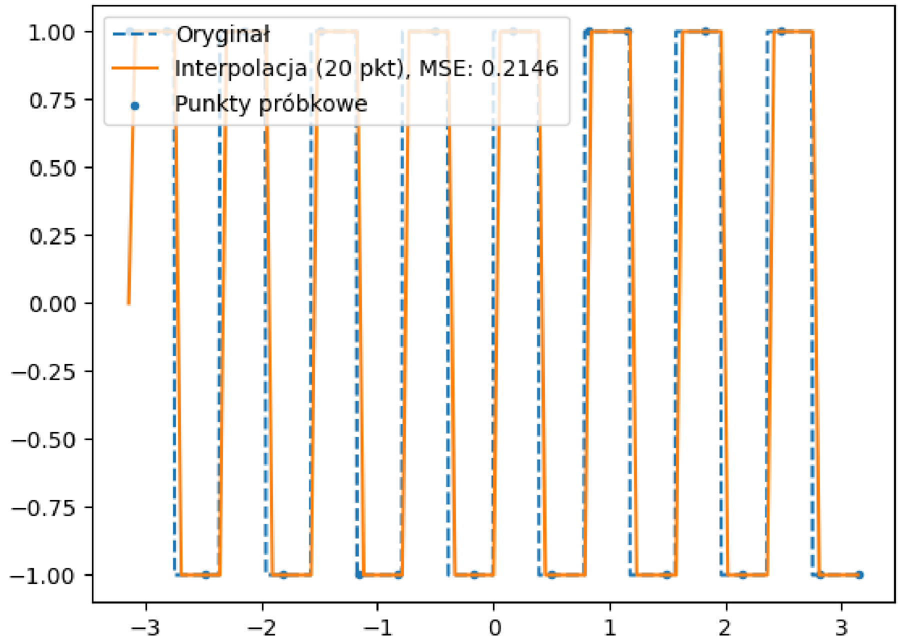
Rysunek 24 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 100 punktów próbkowania, kryterium MSE=0.0486

c) $\boldsymbol{f_3(x) = sgn(\sin(8x))}$

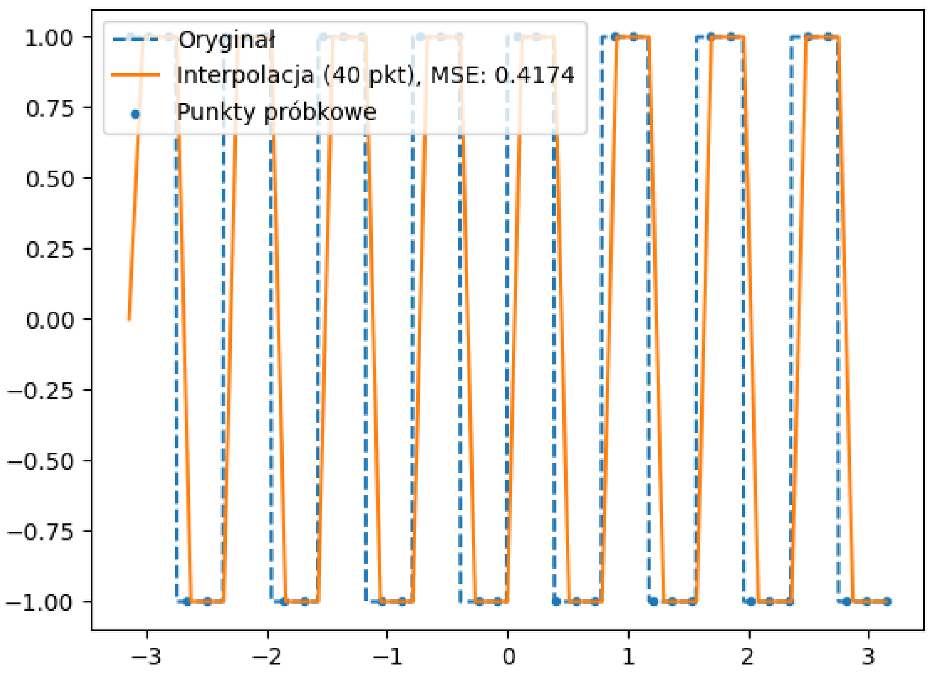
- Jądro $h_1(x)$



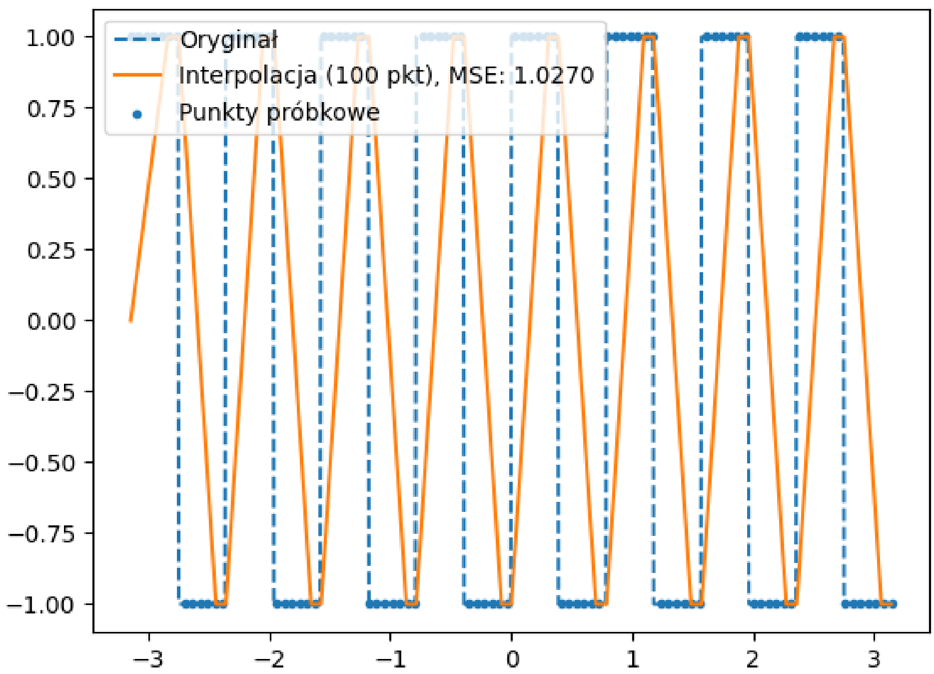
Rysunek 25 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 10 punktów próbkowania, kryterium MSE=0.1144



Rysunek 26 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 20 punktów próbkowania, kryterium MSE=0.2146

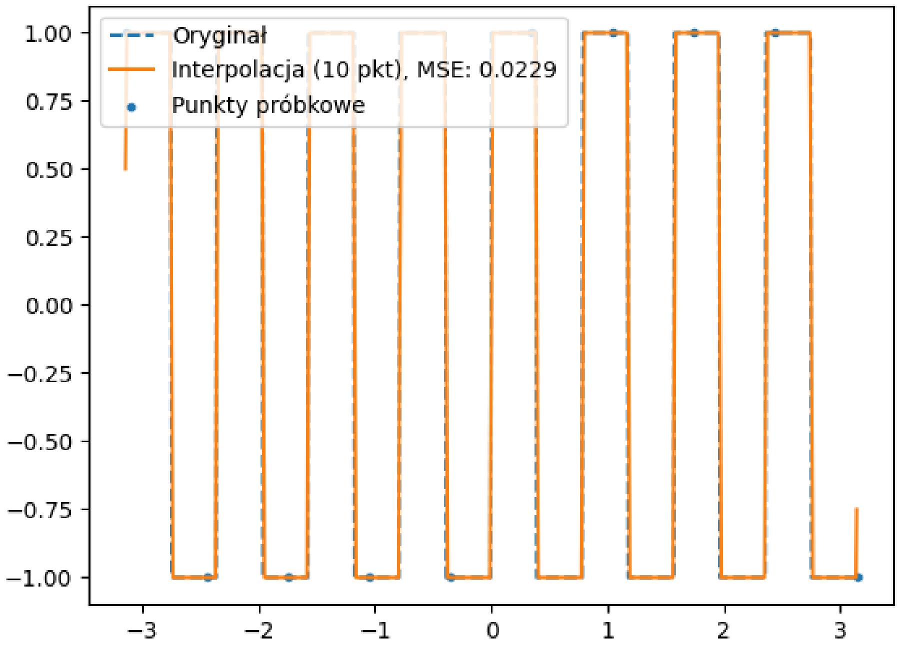


Rysunek 27 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 40 punktów próbkowania, kryterium MSE=0.4174

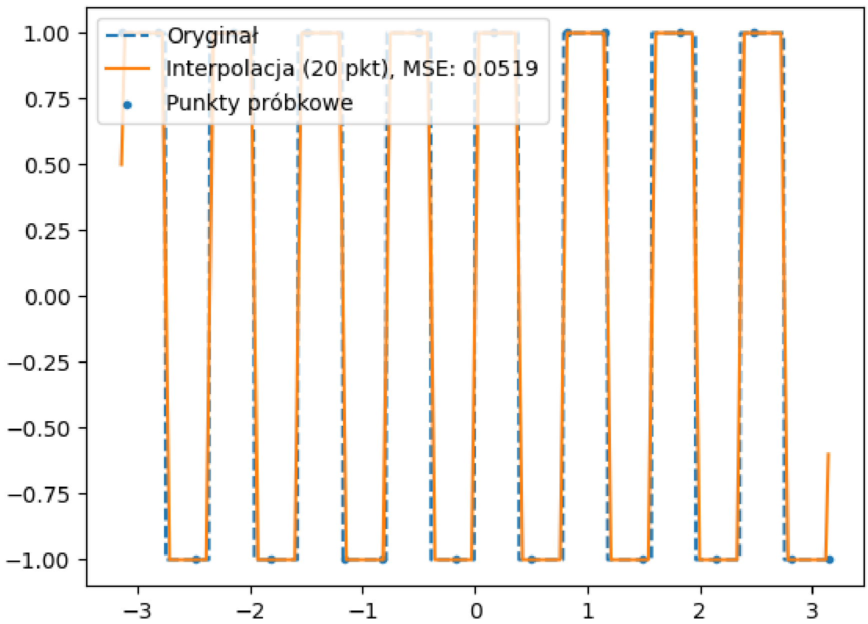


Rysunek 28 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_1(x)$ dla 100 punktów próbkowania, kryterium MSE=1.0270

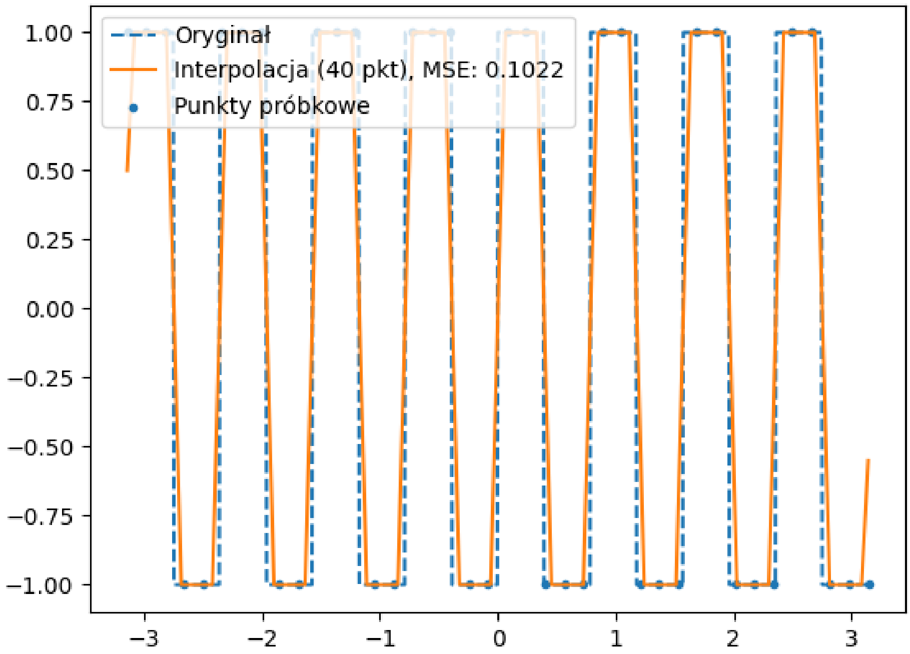
- Jądro $h_2(x)$



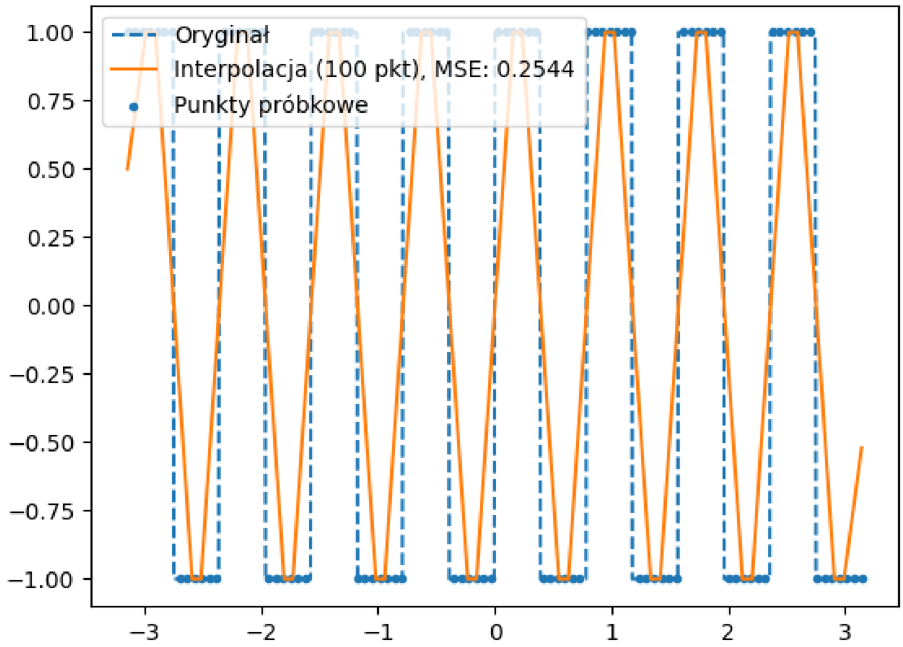
Rysunek 29 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 10 punktów próbkowania, kryterium MSE=0.0229



Rysunek 30 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 20 punktów próbkowania, kryterium MSE=0.0519

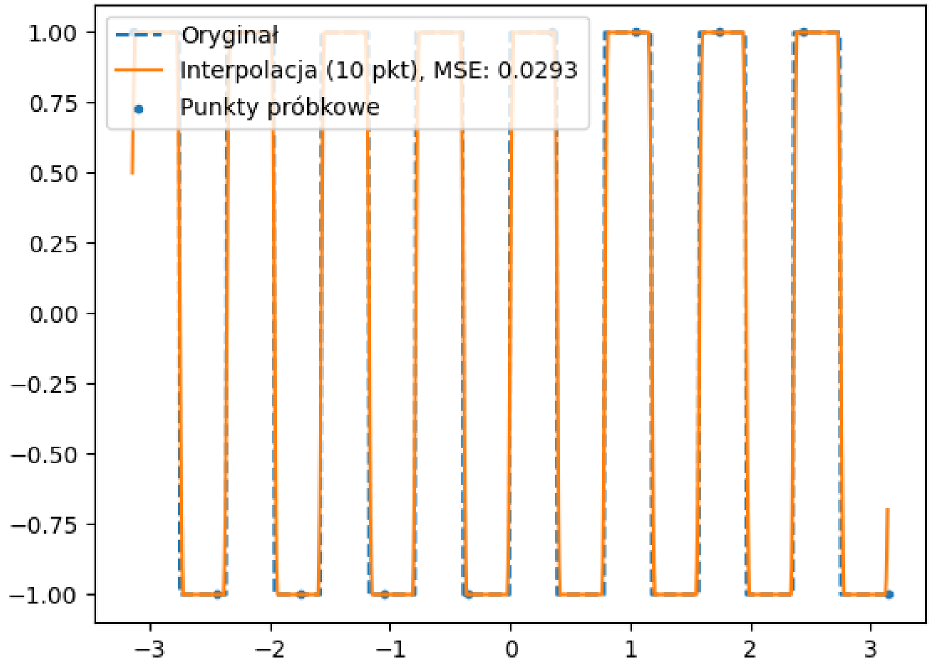


Rysunek 31 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 40 punktów próbkowania, kryterium MSE=0.1022

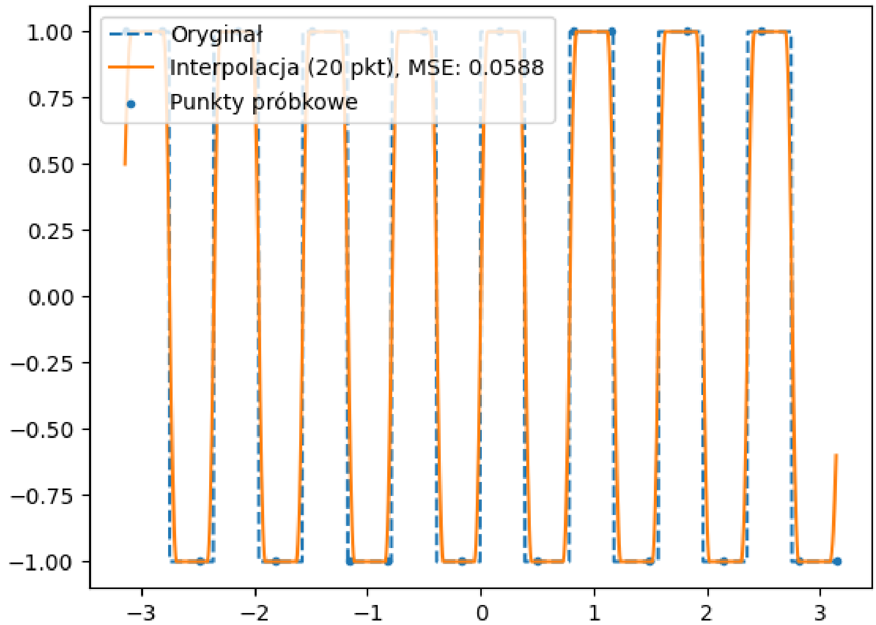


Rysunek 32 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_2(x)$ dla 100 punktów próbkowania, kryterium MSE=0.2544

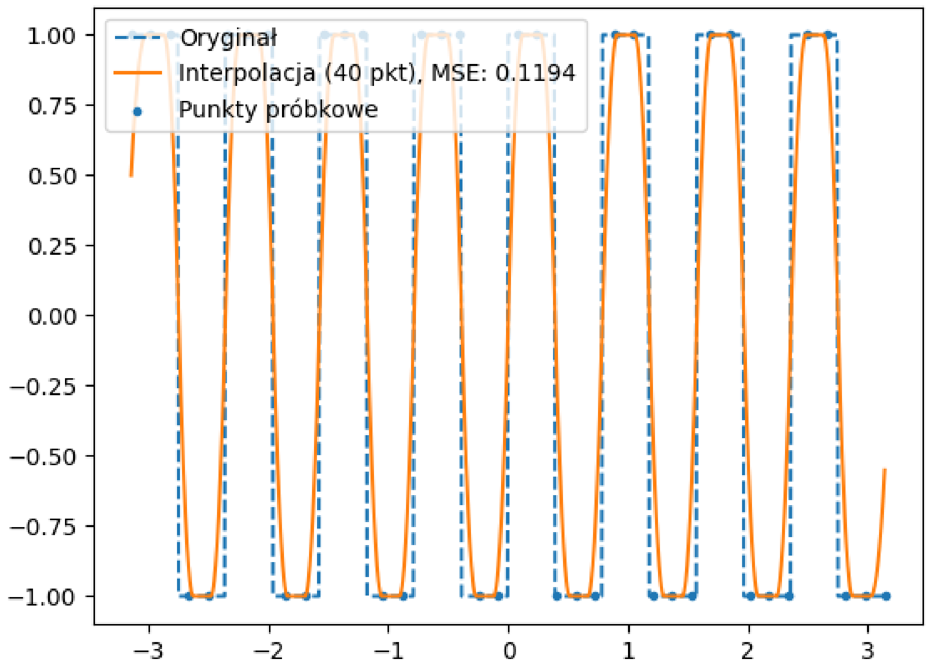
- Jądro $h_3(x)$



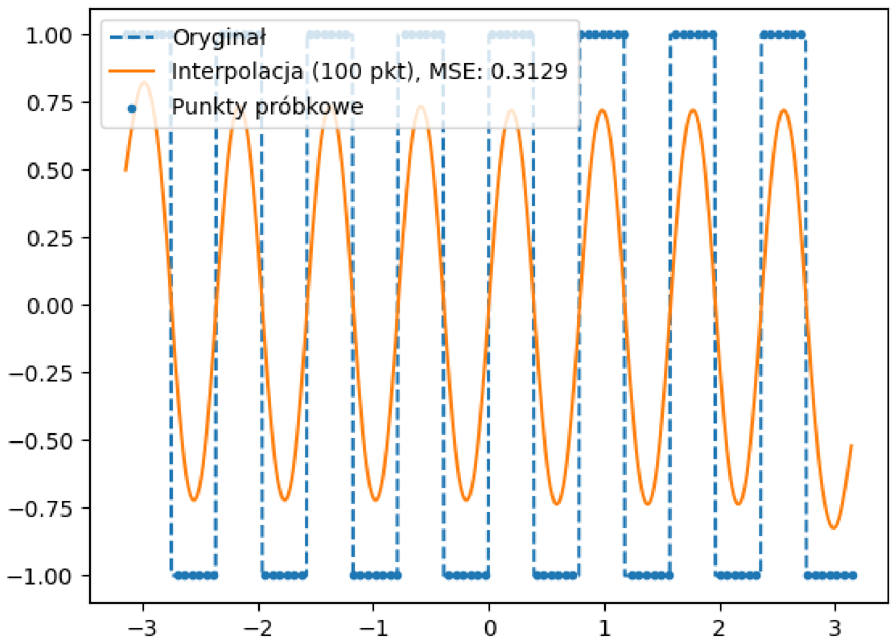
Rysunek 33 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 10 punktów próbkowania, kryterium MSE=0.0293



Rysunek 34 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 20 punktów próbkowania, kryterium MSE=0.0588



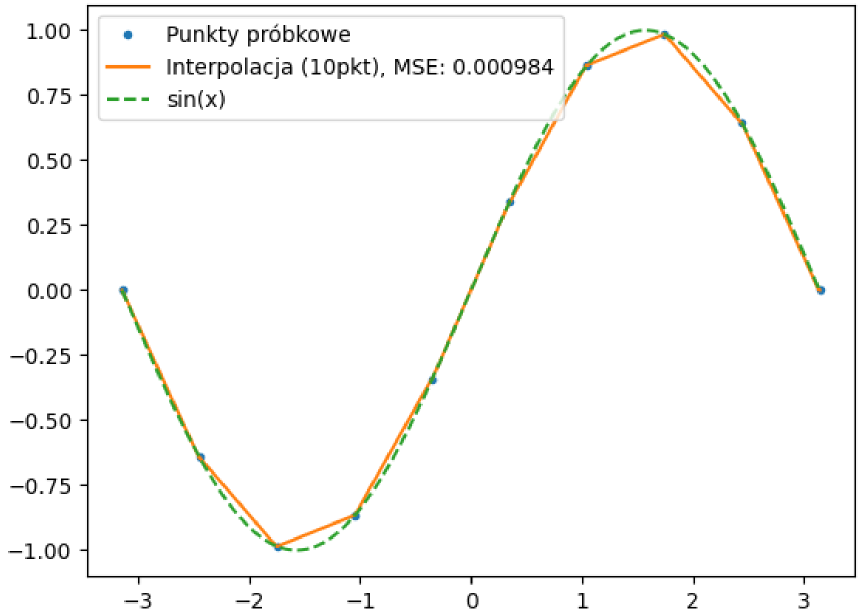
Rysunek 35 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 40 punktów próbkowania, kryterium MSE=0.1194



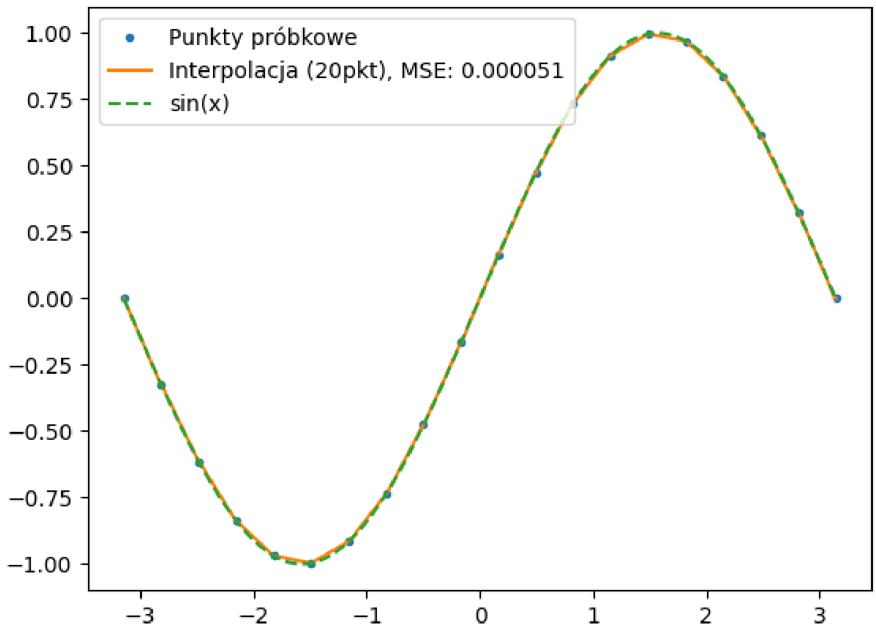
Rysunek 36 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną z pomocą konwolucji jądrem $h_3(x)$ dla 100 punktów próbkowania, kryterium MSE=0.3129

II. Interpolacja średnią

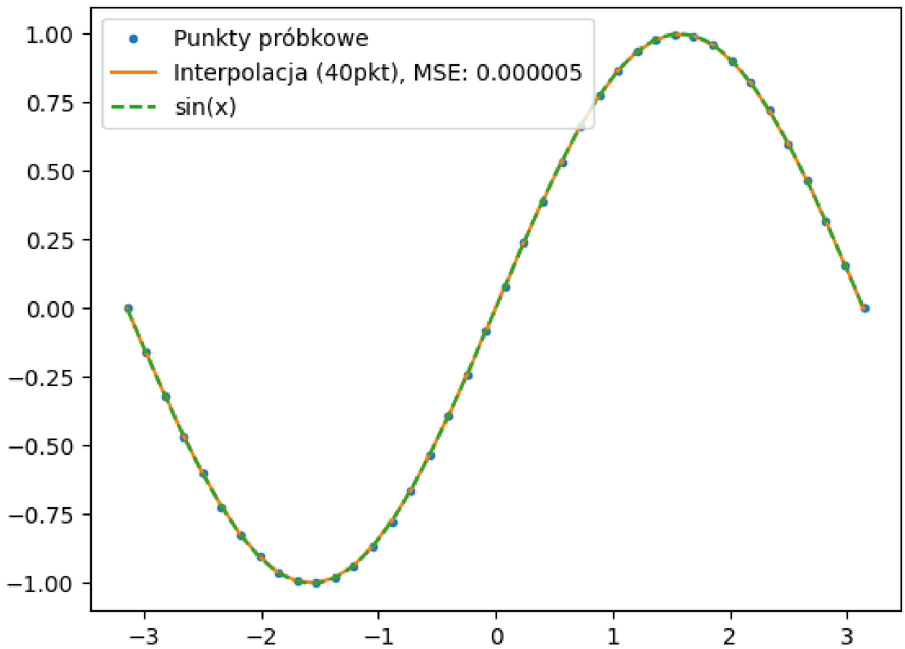
a) $\boldsymbol{f_1(x) = \sin(x)}$



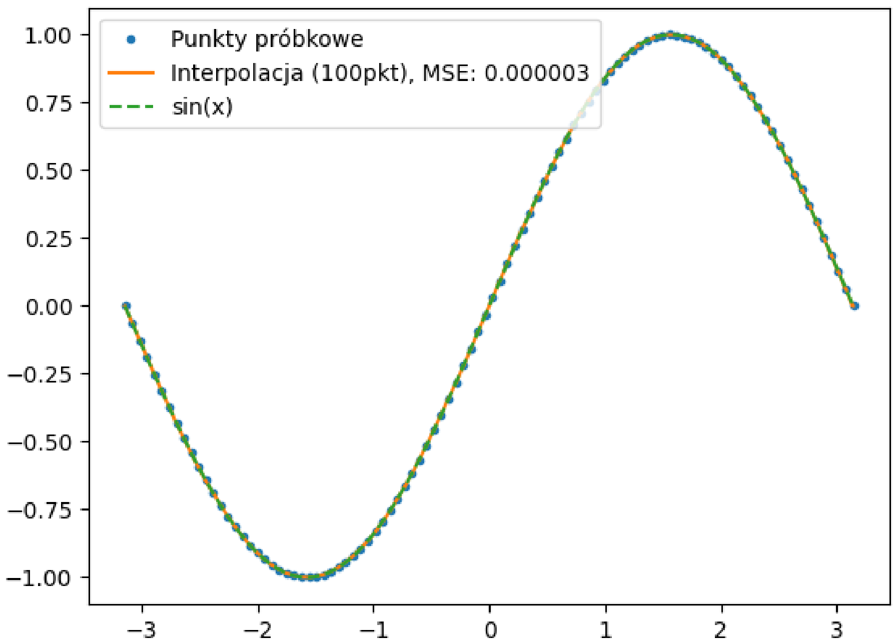
Rysunek 37 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 10 punktów próbkowania, kryterium MSE=0.000984



Rysunek 38 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 20 punktów próbkowania, kryterium MSE=0.000051

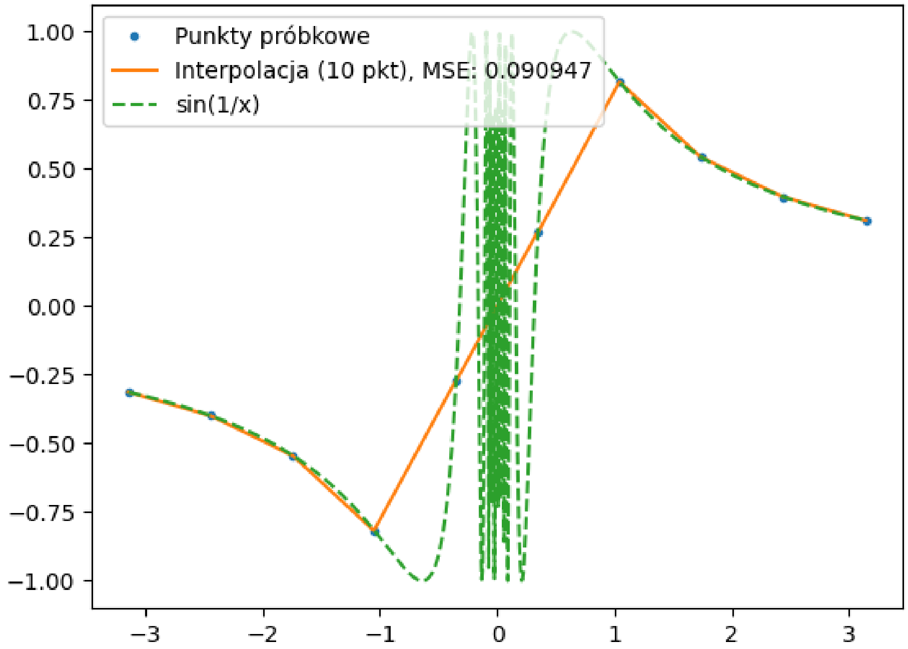


Rysunek 39 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 40 punktów próbkowania, kryterium MSE=0.000005

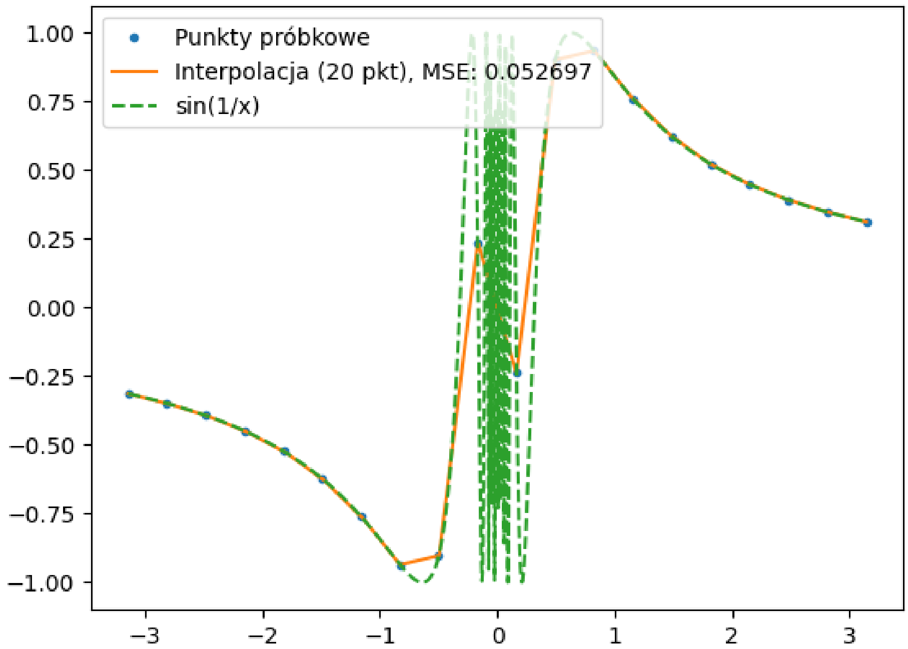


Rysunek 40 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 100 punktów próbkowania,
kryterium MSE=0.000003

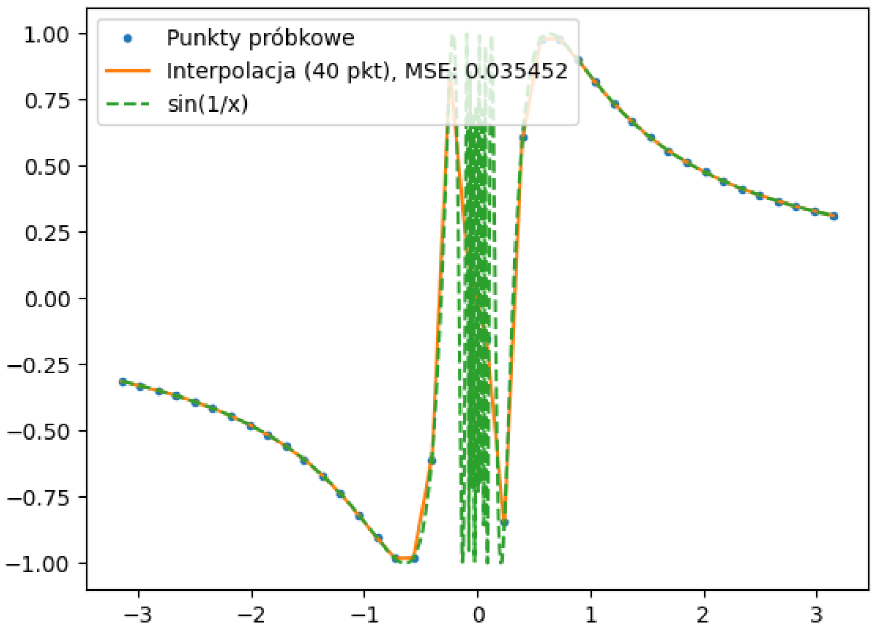
b) $\boldsymbol{f_2(x) = \sin(x^{-1})}$



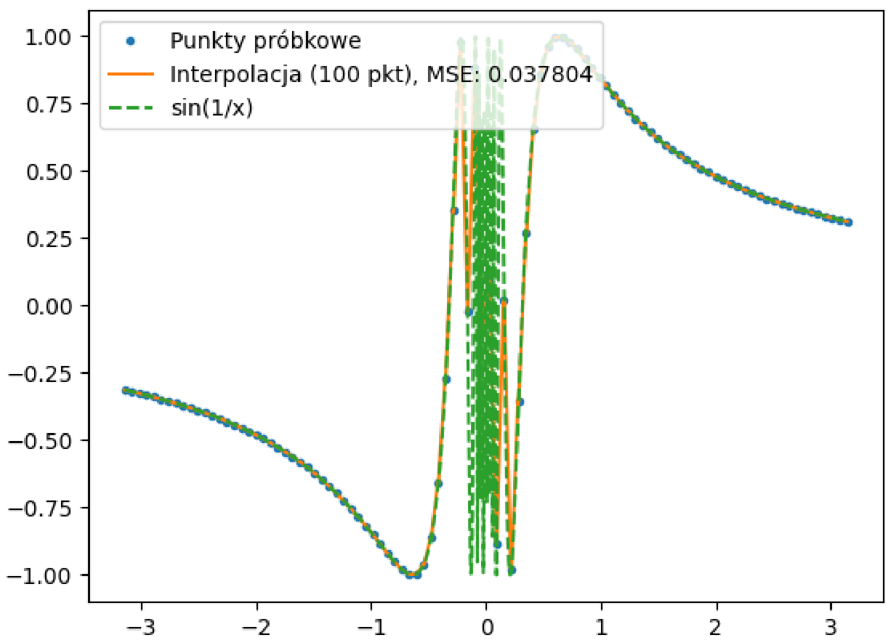
Rysunek 41 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną dla 10 punktów próbkowania, kryterium MSE=0.090947



Rysunek 42 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną dla 20 punktów próbkowania, kryterium MSE=0.052697

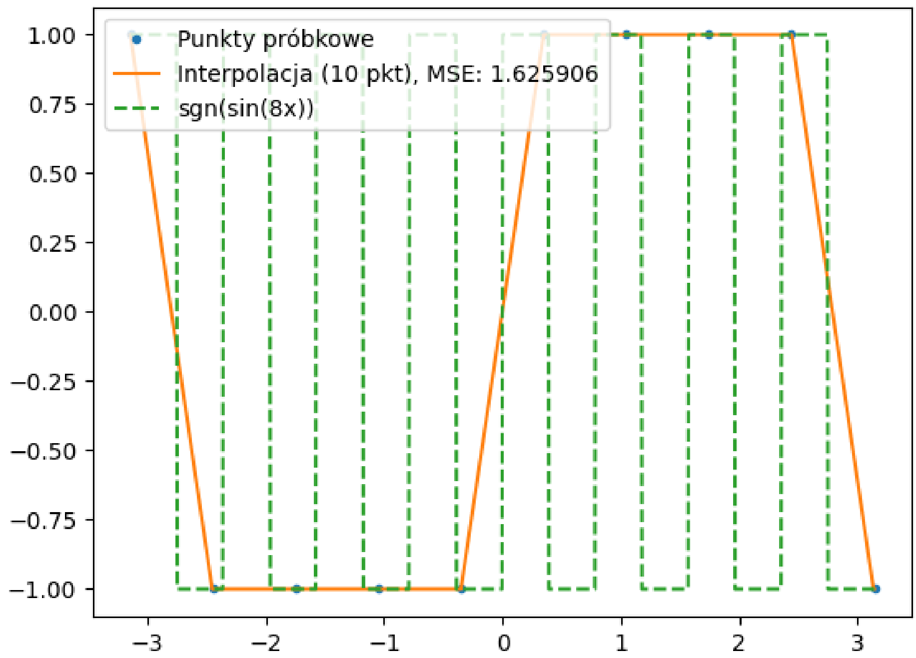


Rysunek 43 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną dla 40 punktów próbkowania, kryterium MSE=0.035452

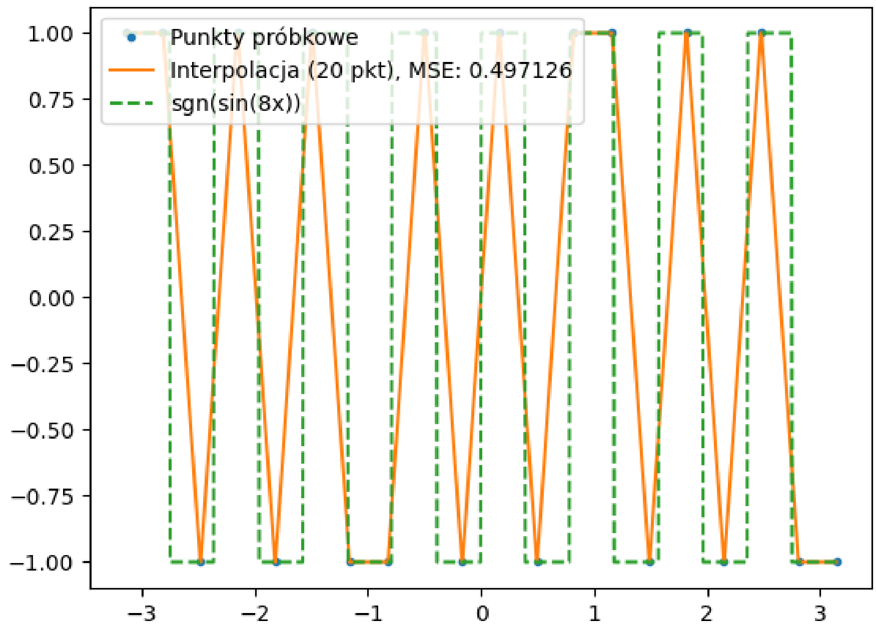


Rysunek 44 wykres przedstawiający funkcje oryginalną $f_2(x)$ oraz interpolowaną dla 100 punktów próbkowania,
kryterium MSE=0.037804

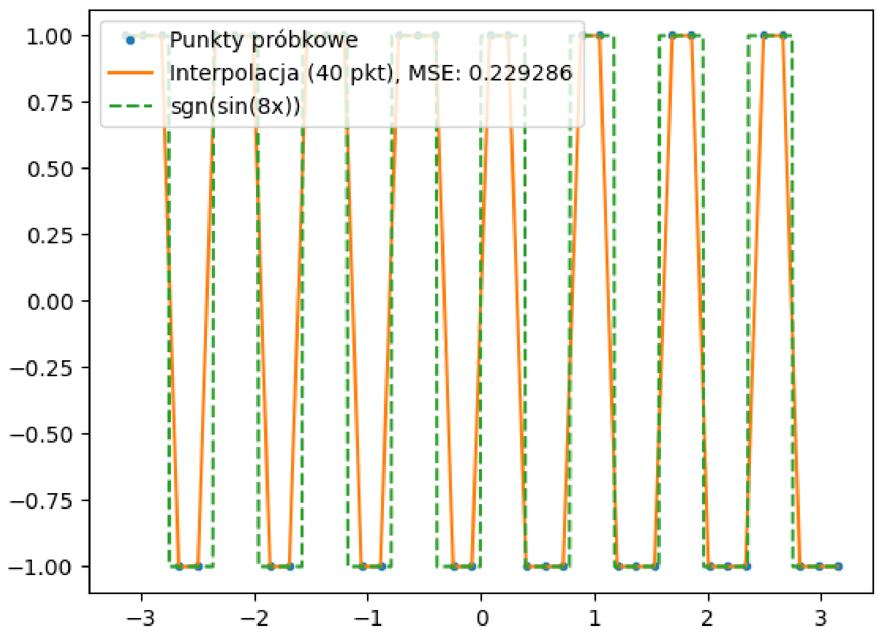
c) $f_3(x) = sgn(\sin(8x))$



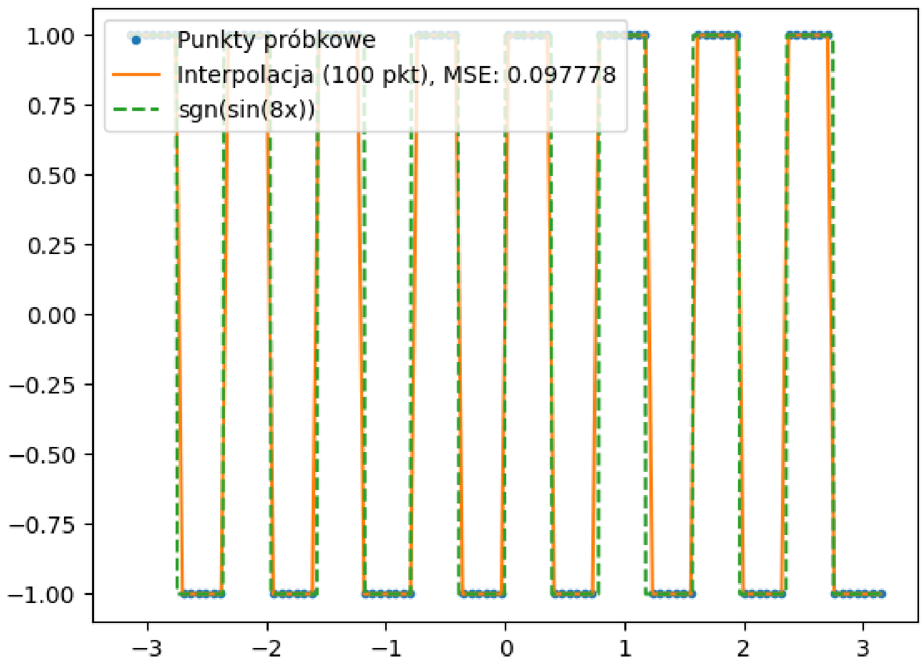
Rysunek 45 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną dla 10 punktów próbkowania, kryterium MSE=1.625906



Rysunek 46 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną dla 20 punktów próbkowania, kryterium MSE=0.497126



Rysunek 47 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną dla 40 punktów próbkowania, kryterium MSE=0.229286

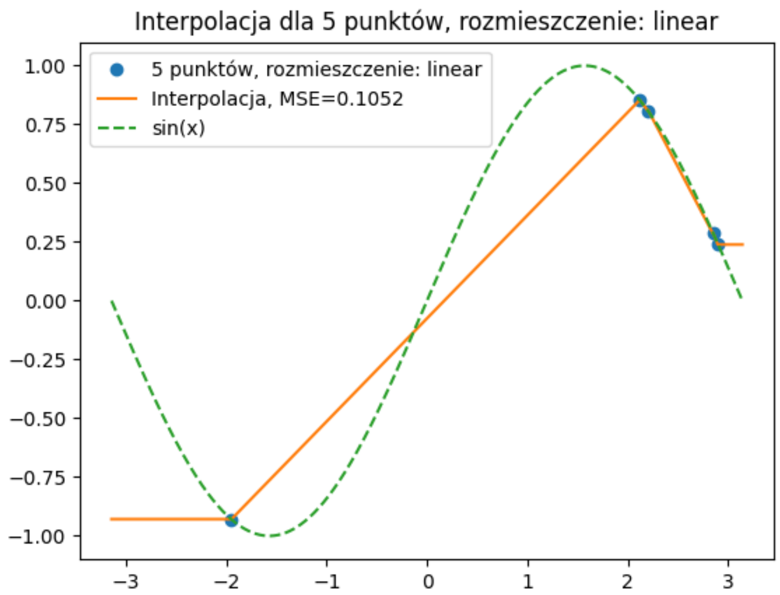


Rysunek 48 wykres przedstawiający funkcje oryginalną $f_3(x)$ oraz interpolowaną dla 100 punktów próbkowania,
kryterium MSE=0.097778

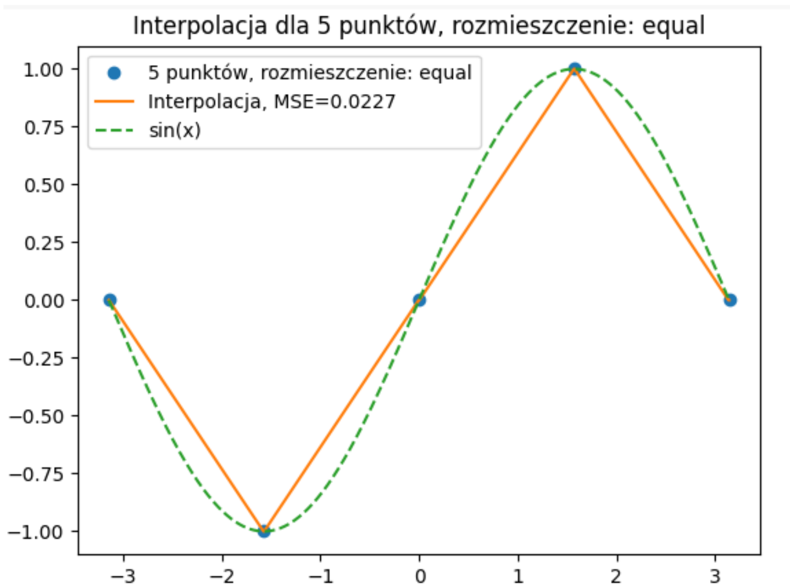
III. Rozmieszczenie

Możemy również wybrać jak chcemy by rozkładały się punkty oraz w jakiej ilości i to także zmienia znacząco wynik interpolacji.

- **5 punktów**

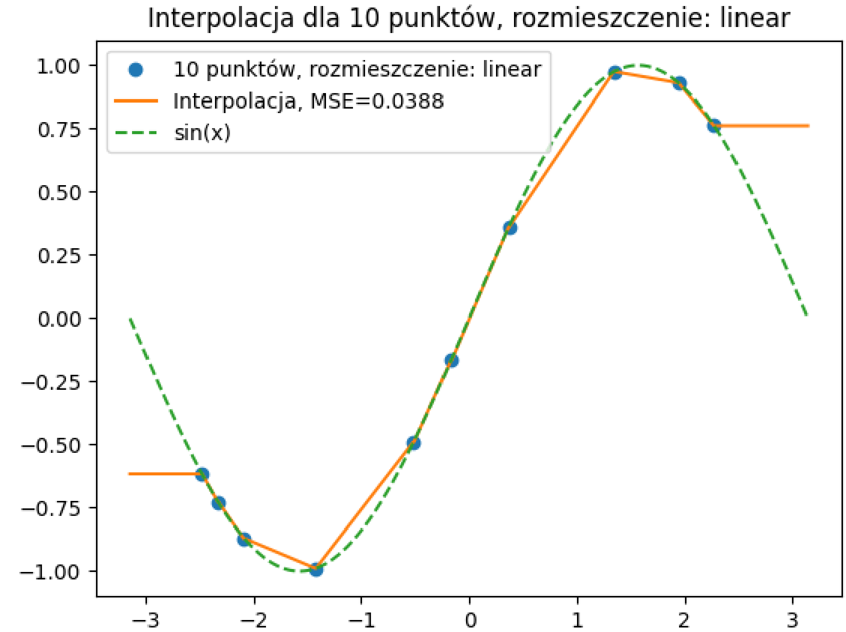


Rysunek 49 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 5 punktów próbkowania z rozmieszczeniem linear,
kryterium MSE=0.1052

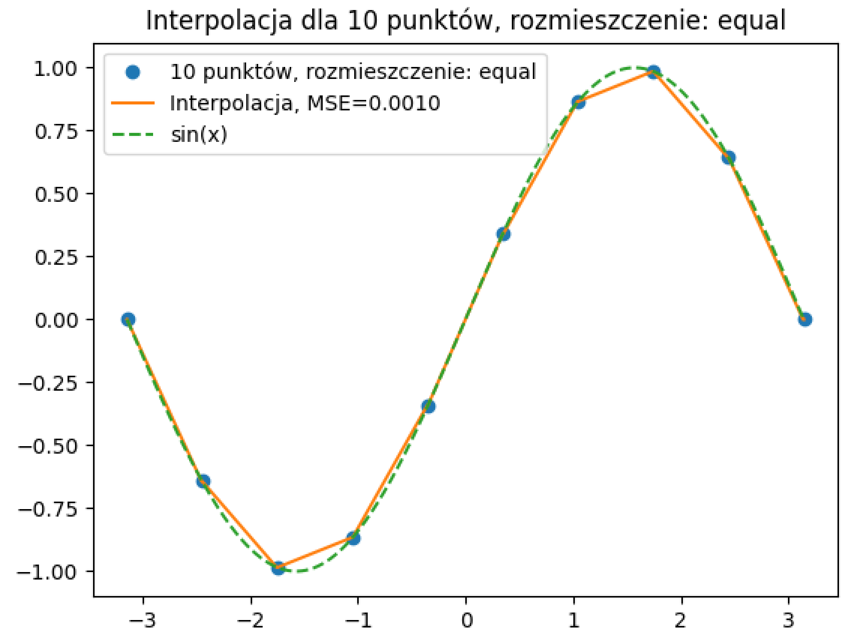


Rysunek 50 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 5 punktów próbkowania z rozmieszczeniem equal,
kryterium MSE=0.0227

- **10 punktów**

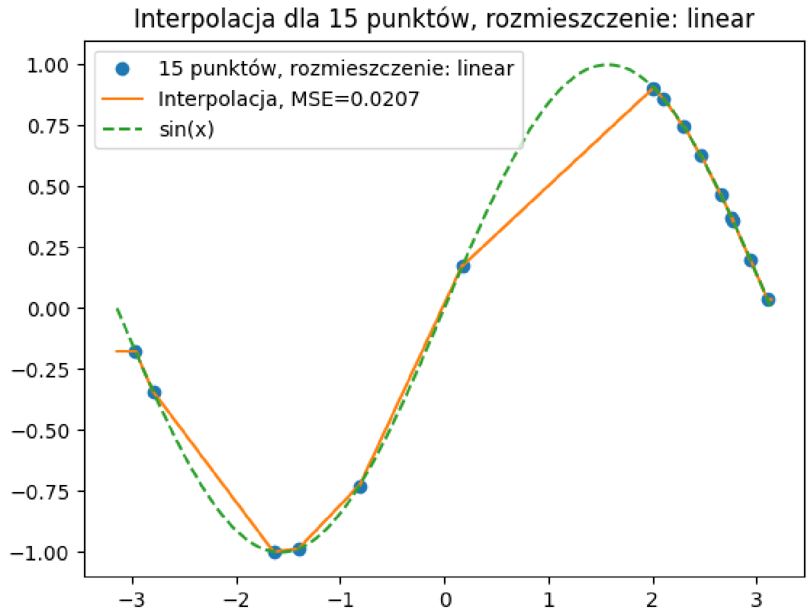


Rysunek 51 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 10 punktów próbkowania z rozmieszczeniem linear,
kryterium MSE=0.0388

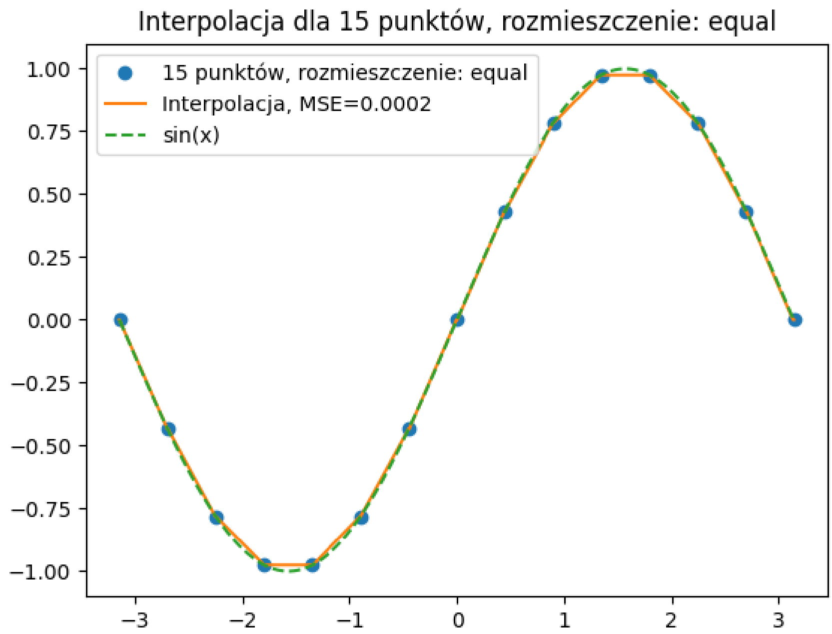


Rysunek 52 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 10 punktów próbkowania z rozmieszczeniem equal,
kryterium MSE=0.0010

- **15 punktów**

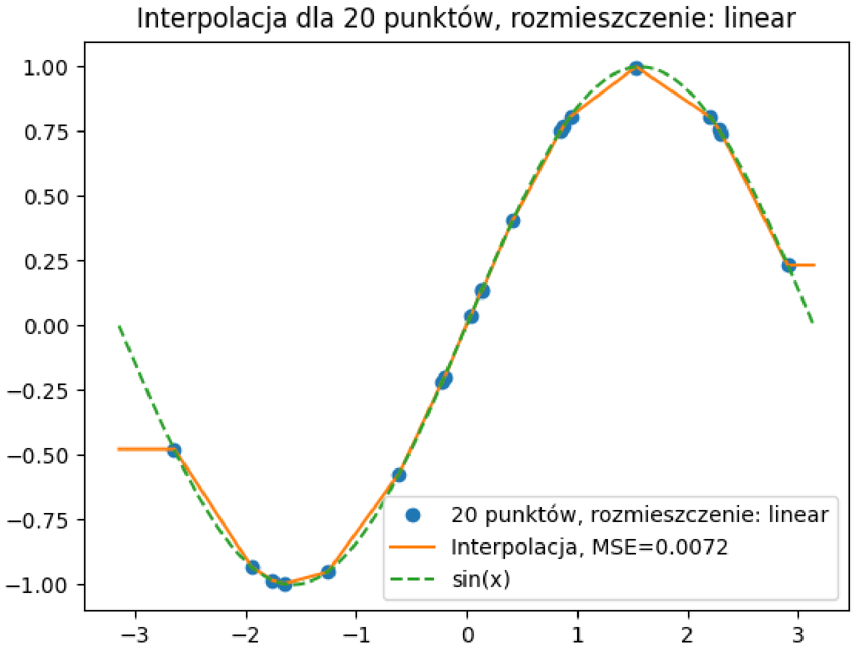


Rysunek 53 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 15 punktów próbkowania z rozmieszczeniem linear,
kryterium MSE=0.0207

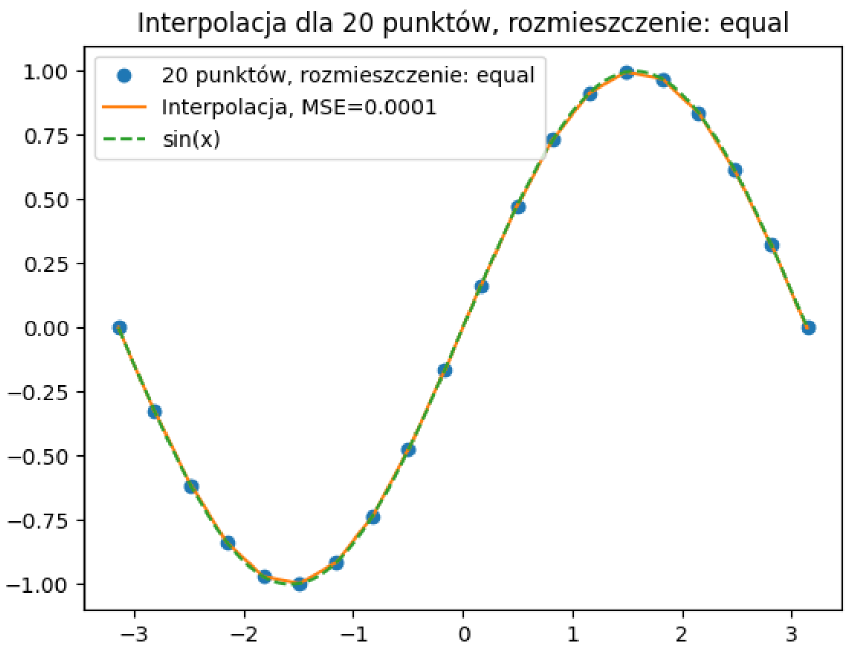


Rysunek 54 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 15 punktów próbkowania z rozmieszczeniem equal,
kryterium MSE=0.0002

- **20 punktów**



Rysunek 55 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 20 punktów próbkowania z rozmieszczeniem linear,
kryterium MSE=0.0072



Rysunek 56 wykres przedstawiający funkcje oryginalną $f_1(x)$ oraz interpolowaną dla 20 punktów próbkowania z rozmieszczeniem equal,
kryterium MSE=0.0001

4. Wnioski

- Istnieje wiele metod interpolacji każda z nich działa w nieco odmienny sposób i charakteryzuje się innymi cechami. Dobór odpowiedniej interpolacji do funkcji jest więc kluczowy jako że nie każda metoda interpolacji sprawdzi się tak samo dobrze jak inna co widać na powyższych wykresach gdzie czasem dochodzi do zniekształceń przez zbyt nagłą zmianę wartości argumentów funkcji.
- Tak samo jak kluczowy jest dobór interpolacji tak samo jest kluczowy dobór jądra do konwolucji i należy wybrać odpowiednie w zależności od potrzeb sytuacji.
- Im więcej punktów interpolacji tym lepiej i prowadzi to do lepszej jakości. Im więcej punktów, tym dokładniejsza reprezentacja oryginalnej funkcji, i jest w stanie przedstawić jej szczegóły. Z kolei z mniejszą liczbą punktów interpolacji tracimy na dokładności. Interpolacja może być bardziej przybliżona, a istotne drobnostki mogą zostać po prostu pominięte. Trzeba jednak znów pamiętać że bardzo zależy to od funkcji i powyżej mogliśmy zobaczyć że nie zawsze większa ilość punktów oznacza lepsza jakość.
- Jednakże nie tylko ilość punktów jest ważna ale też ich rozkład. Jeśli punkty interpolacji są równomiernie rozłożone, to jest to dobry sposób na ogólne reprezentowanie większości funkcji (choć nie zawsze, jako że bywają wyjątki gdzie funkcja może nie zawsze zachowywać się regularnie). Szczególnie gdy funkcja jest gładka, równomierne rozmieszczenie punktów pozwoli nam skutecznie przedstawić wykres. Jednakże należy pamiętać że równomierne rozłożenie punktów w niektórych przypadkach może czasem prowadzić do aliasingu.
- W niektórych przypadkach jednak, nierównomierne rozmieszczenie punktów może być bardziej efektywne, szczególnie w obszarach o dużych zmianach funkcji. Może to pomóc w uchwyceniu lokalnych szczegółów i szybkich zmian na krótkim odcinku.
- Ostateczny wpływ zależy od konkretnego przypadku i charakterystyki funkcji. Wybór liczby punktów i ich rozmieszczenia zależy od tego, co chcemy uzyskać i jakie są właściwości badanej funkcji. Jak widzimy każda metoda ma swoje ograniczenia. Interpolacja przez konwolucje sprawdzi się dla prostych funkcji o pewnych regularnościach i ograniczonej liczbie gwałtownych zmian lecz będzie nieskuteczna dla bardziej złożonych funkcji o nieregularnych kształtach i zmianach (no chyba że bardzo odpowiednio ją dostosujemy). Co mogliśmy zaobserwować powyżej gdzie niekoniecznie pokrywała się z oryginalnymi funkcjami bardziej zaawansowanymi i wychodziła spora różnica w kryterium MSE.

5. Cel ćwiczenia #2- Skalowanie Obrazów

Celem tego ćwiczenia jest zrozumienie i praktyczne zastosowanie algorytmów skalowania obrazów cyfrowych oraz ich wpływu na jakość otrzymanego obrazu. Przede wszystkim zadanie opiera się na implementacji algorytmu skalowania obrazów cyfrowych z pomocą interpolacji funkcji czy operacji splotu dyskretnego i zastosowanie różnorakich metod zmniejszania, powiększania oraz zebranie wniosków na temat efektywności, wadach i zaletach każdej z nich.

6. Wstęp teoretyczny

➢ Skalowanie obrazów na podstawie interpolacji

Interpolacja, w tym kontekście, to metoda estymacji nowych wartości pikseli, aby uzyskać obraz o innej rozdzielczości niż oryginał. Skalowanie obrazu może być stosowane w różnych sytuacjach, takich jak powiększanie, zmniejszanie lub dostosowywanie proporcji.

a) Zmniejszanie przez splot z jądrem uśredniającym

Operacja zmniejszania obrazu za pomocą splotu z jądrem uśredniającym polega na zastosowaniu filtru (jądra) uśredniającego. Przy zastosowaniu splotu wynikowy obraz będzie zawierać średnią wartości pikseli z danego rejonu obrazu, w zależności od rozmiaru jądra które oblicza średnią wartość pikseli w określonym obszarze. To działanie powoduje, że obraz staje się mniej szczegółowy, co skutkuje zmniejszeniem rozmiaru obrazu.

$$K_m = \frac{1}{N}\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}$$

b) Algorytm pomniejszania average/max pooling

Algorytm pomniejszania obrazu, używając operacji średniej z obszaru (average pooling) lub max pooling, polega na dzieleniu obrazu na mniejsze obszary i zastępowaniu każdego obszaru jednym pikselem. W przypadku average pooling, nowa wartość piksela jest obliczana jako średnia wartość pikseli w danym obszarze, natomiast w max pooling, nowa wartość to maksymalna wartość w danym obszarze.

c) Interpolacacja dwuliniowa

Interpolacja dwuliniowa (bilinear interpolation) to technika stosowana w grafice komputerowej i przetwarzaniu obrazów do wyznaczania wartości pikseli w nowej lokalizacji pomiędzy już znanymi pikselami w oryginalnym obrazie. Jest to jedna z metod interpolacji, która zakłada, że wartości pikseli zmieniają się liniowo wzdłuż obu osi poziomej i pionowej. Podczas interpolacji dwuliniowej nowa wartość piksela obliczana jest na podstawie otaczających go czterech pikseli w oryginalnym obrazie. Zakłada się, że zmiana wartości piksela jest liniowa w obu kierunkach, dlatego używa się dwóch kierunków interpolacji.

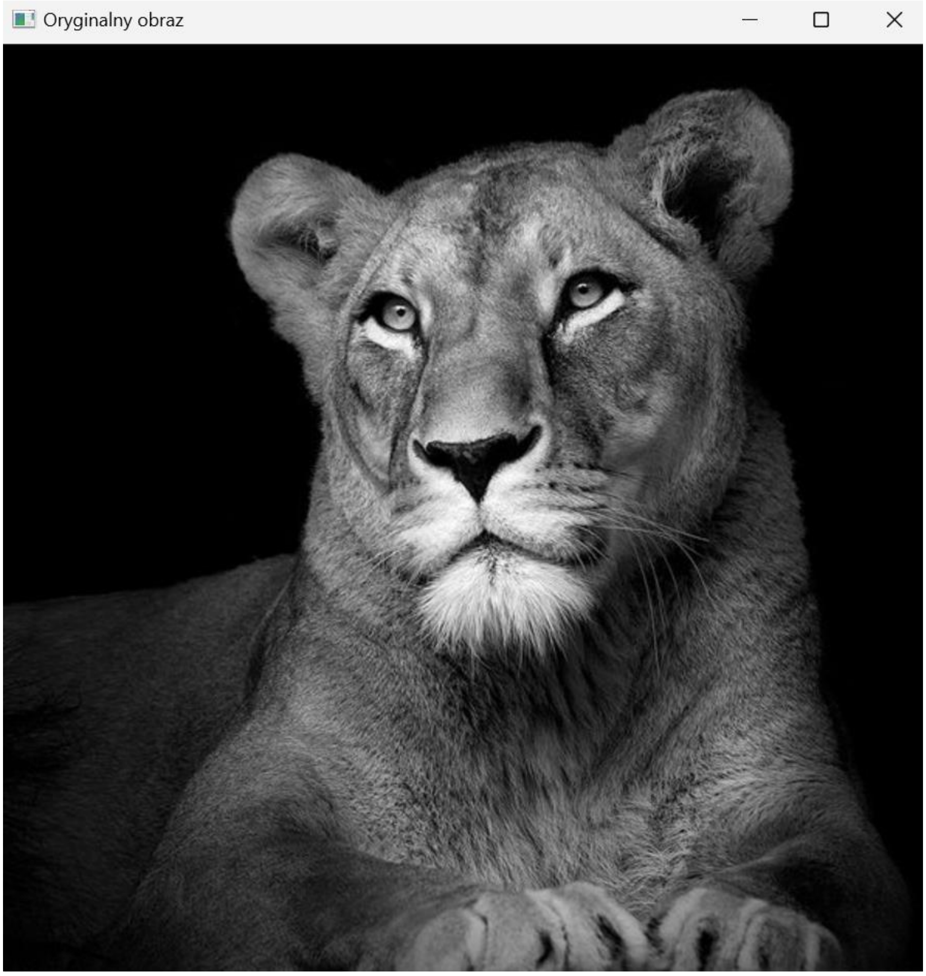
d) Interpolacja najbliższym sąsiadem

W kontekście skalowania obrazu, interpolacja najbliższego sąsiada jest szybką i łatwą do zrozumienia metodą. Interpolacja najbliższego sąsiada to prosty algorytm interpolacji, który wybiera wartość piksela z oryginalnego obrazu najbliższego piksela w nowym obrazie. Oznacza to, że dla każdego piksela w nowym obrazie, algorytm wybiera wartość z oryginalnego obrazu, która znajduje się najbliżej tego piksela. Jednak może wprowadzać efekt schodkowania (aliasing), szczególnie przy powiększaniu obrazów, ponieważ nowe piksele mogą mieć wartości, które są skokami od pikseli oryginalnego obrazu.

7. Przebieg ćwiczenia

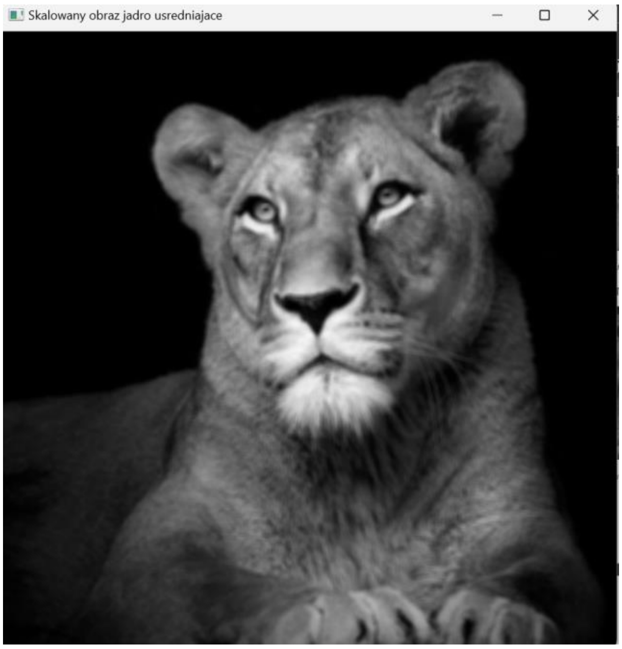
Zadanie polegało na zaimplementowaniu algorytmu, który umożliwi zmniejszanie i powiększanie obrazów przy użyciu różnych metod interpolacji funkcji oraz operacji splotu dyskretnego. Najpierw, na podstawie splotu z jądrem uśredniającym, należało wykonać operację zmniejszania obrazu. Następnie trzeba było zaimplementować powiększanie obrazu za pomocą 2 interpolacji funkcji, korzystając z wcześniejszego ćwiczenia. Po czym ocenianić jakość algorytmów zarówno, poprzez wizualną ocenę wyników, ale też zastosowanie miary MSE do porównania oryginalnego i przetworzonego obrazu. Dodatkowo można było zaimplementować inne algorytmy pomniejszania obrazu (np. max pooling), operacji zmniejszania i zwiększania o niecałkowitą wielokrotność oraz mogliśmy porównać skalowanie obrazu z sekwencją mniejszych obrazów. Istniała także opcja rozbudowy implementacji na kolorowe obrazy.

a) Operacje zmniejszania



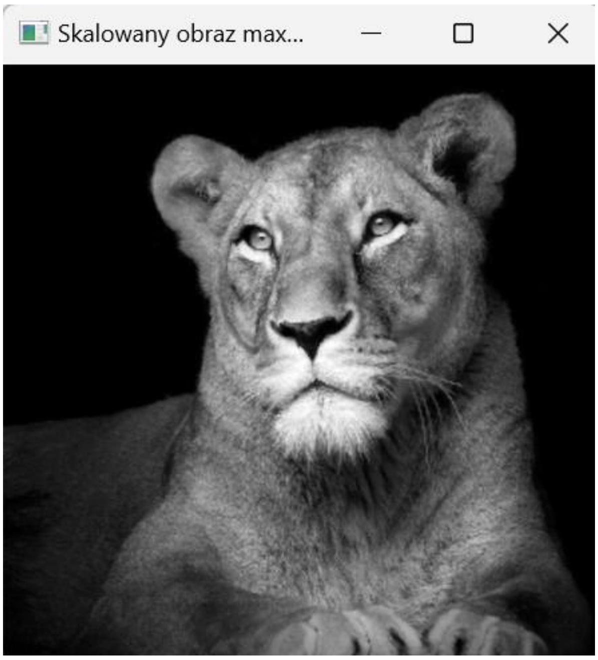
Rysunek 57 Obraz czarnobiały oryginalny

➢ Splot z jądrem uśredniającym



Rysunek 58 Obraz czarnobiały przeskalowany metodą splotu z jądrem uśredniającym k=0.5

➢ Algorytm max pooling



Rysunek 59 Obraz czarnobiały przeskalowany metodą max pooling k=0.5

b) operację powiększania obrazu za pomocą dwóch interpolacji funkcji

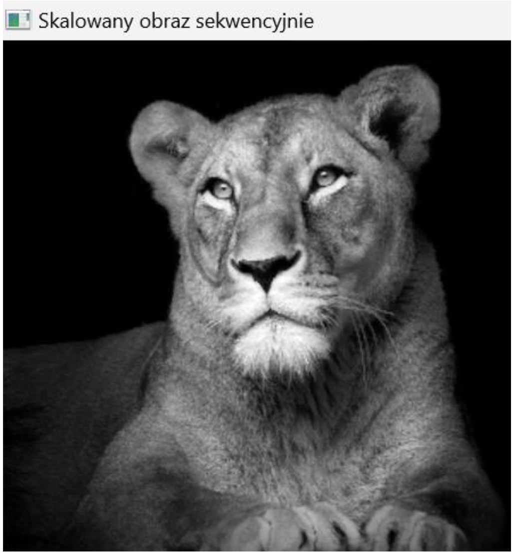
➢ Interpolacja dwuliniowa

Rysunek 60 Obraz czarnobiały przeskalowany interpolacją dwuliniową k=1.5 (jest to jedynie część, cały obraz po operacji nie zmieścił się w ekranie)

- Interpolacja najbliższy sąsiad (Pojawia się Aliasing)

Rysunek 61 Obraz czarnobiały przeskalowany interpolacją najbliższego sąsiada k=1.5 (jest to jedynie część, cały obraz po operacji nie zmieścił się w ekranie)

c) Porównanie skalowania obrazu ze skalowaniem obrazu z sekwencją mniejszych



Rysunek 62 Obraz czarnobiały przeskalowany sekwencją mniejszych

Rysunek 63 Obraz czarnobiały przeskalowany bezpośrednio

Gołym okiem jesteśmy w stanie zauważyć wyraźną różnice między wersją powiększoną, pomniejszoną a oryginałem. Przy obu tych operacjach tracimy na jakości zdjęcia i robi się nie wyraźne co potwierdza poniższa metryka MSE. W zależności od wybranej metody otrzymamy różne rezultaty rozmycia i obrazy będą różnić się szczegółami i rozkładem pikseli. Obraz skalowany sekwencyjnie jest bardzo podobny jednakże wydaje się być nieco mniej szczegółowy niż bezpośrednio co także potwierdza kryterium MSE.

```
MSE od oryginału do pomniejszonego: 41.0537
MSE od oryginału do powiększonego: 78.1330
MSE bezposrednio: 76.1152
MSE sekwencyjnie: 76.2563
```

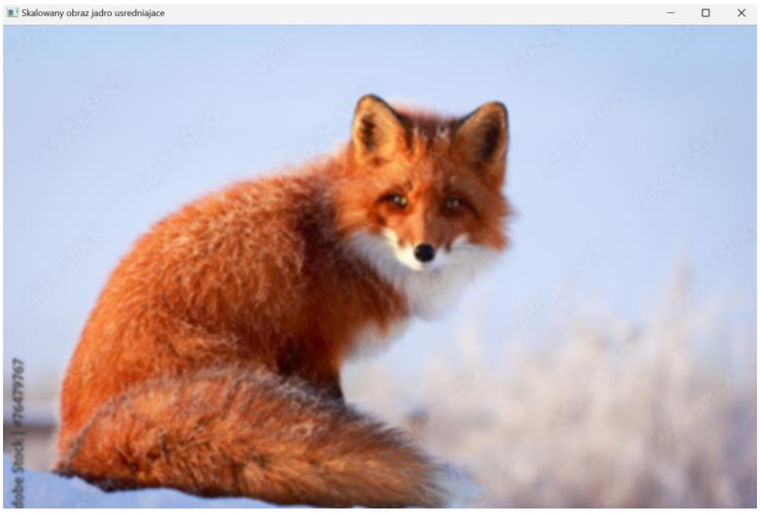
Rysunek 64 kryteria MSE pomniejszonego oraz powiększonego zdjęcia w konsoli

Kolorowe obrazy

d) Operacje zmniejszania

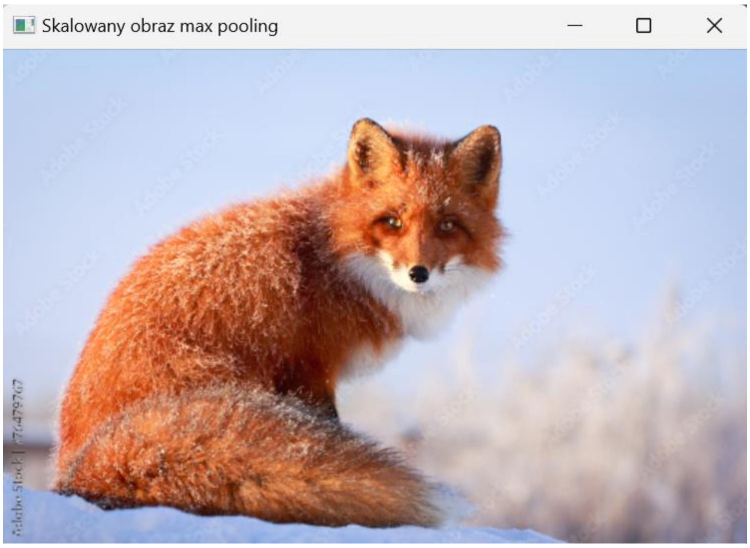
Rysunek 65 obraz kolorowy oryginalny

- Splot z jądrem uśredniającym



Rysunek 66 obraz kolorowy przeskalowany jądrem uśredniającym k=0.5

- Algorytm max pooling



Rysunek 67 Obraz kolorowy przeskalowany metodą max pooling k=0.5

e) operację powiększania obrazu za pomocą dwóch interpolacji funkcji

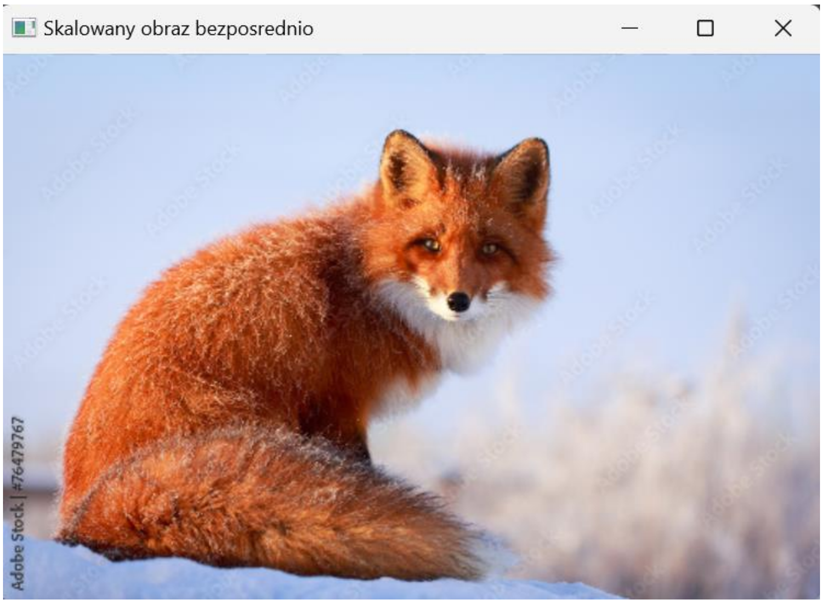
- Interpolacja dwuliniowa

Rysunek 68 Obraz kolorowy przeskalowany interpolacją dwuliniową k=1.5 (jest to jedynie część, cały obraz po operacji nie zmieścił się w ekranie)

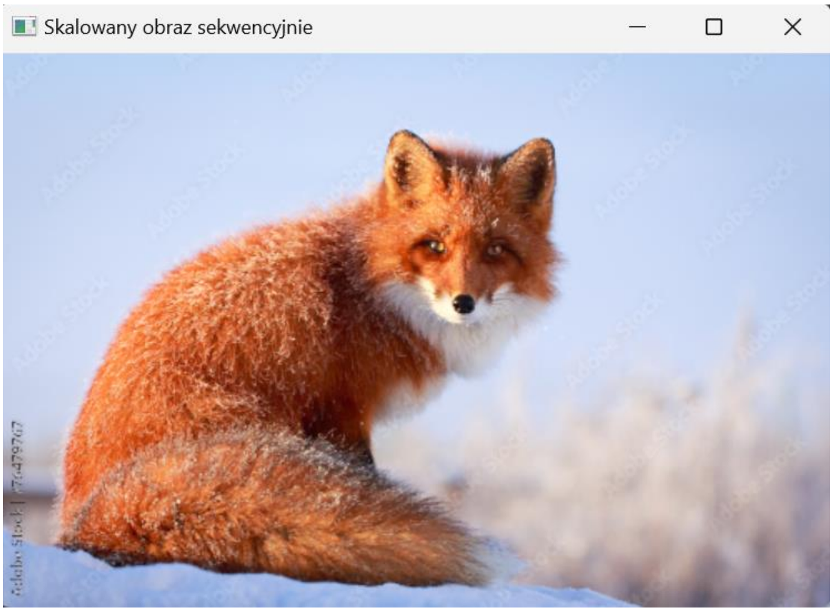
- Interpolacja najbliższy sąsiad (Pojawia się Aliasing)

Rysunek 69 Obraz kolorowy przeskalowany interpolacją najbliższego sąsiada k=1.5 (jest to jedynie część, cały obraz po operacji nie zmieścił się w ekranie)

f) Porównanie skalowania obrazu ze skalowaniem obrazu z sekwencją mniejszych



Rysunek 70 Obraz kolorowy przeskalowany bezpośrednio



Rysunek 71 Obraz kolorowy przeskalowany sekwencją mniejszych

```
MSE od oryginału do pomniejszonego: 22.8497
MSE od oryginału do powiększonego: 86.6658
MSE bezposrednio: 84.7379
MSE sekwencyjnie: 84.9818
```

Rysunek 72 kryteria MSE pomniejszonego oraz powiększonego zdjęcia wraz z MSE skalowania bezpośredniego oraz sekwencyjnego w konsoli

8. Wnioski

- Obserwujemy wzrost wartości MSE w miarę coraz większego powiększania (czasem też pomniejszania). Wyższe wartości MSE oznaczają większą różnicę między oryginalnym obrazem a obrazem po sekwencji pomniejszania i powiększania czyli im wyższa wartość tego kryterium tym większy błąd. To zrozumiałe, ponieważ powiększenie prowadzi do sztucznego kreowania pikseli po innych pobliskich, tak samo pomniejszenie prowadzi do utraty informacji jako że sprowadzamy obraz do prostszej wersji zajmującej mniej pikseli, co wpływa oczywiście na jakość skalowanego obrazu.
- Analizując wartości MSE, możemy zastanowić się, jak bardzo możemy pomniejszyć w kontekście utraty jakości. Optymalny stopień pomniejszania może być różny w zależności od konkretnego przypadku zastosowania, a analiza wyników MSE pozwoli na wybranie optymalnej wersji. W naszym przypadku obraz na oko nie odstawał mocno jakością lecz metryka MSE wykazała większe wartości dla powiększanego obrazu, co świadczy o tym że jakość powiększonego obrazu o 1.5 była gorsza niż pomniejszonego o 0.5. Sprawdzając dalej okaże się, że im większe pomniejszenie/powiększenie tym gorsza może być jakość obrazu i wraz z kolejnym stopniem skalowania tracimy na detalach.
- Należy uważać ze skalowaniem obrazów zarówno z pomniejszaniem jak i powiększaniem. Skalowanie splotem z jądrem uśredniającym nie jest bardzo problematyczne do zaimplementowania oraz pozwala na zachowanie wielu szczegółów z obrazu oryginalnego jednak i tak może prowadzić do utraty paru istotnych detali. Ta metoda polega również na tym jakiego filtra używamy.
- Powiększanie interpolacją także zachowa wiele informacji z oryginalnego obrazu lecz tu także pojawia się problem z jakością otrzymanego obrazu. Ten często może być nieco rozmyty i przedstawiać lekko zniekształcony obraz w zależności od wielkości skalowania po prostu tak jak w wypadku pomniejszania może zgubić masę drobnych szczegółów.
- Metoda najbliższego sąsiada na pewno jest prostsza od interpolacji dwuliniowej ale jest niedokładna gdyż wykorzystuje jedynie wartość najbliższego piksela, co może prowadzić do utraty informacji oraz może prowadzić do efektu aliasingu!! Co możemy zaobserwować na powyższym obrazie lwa. Więc choć interpolacja dwuliniowa jest bardziej zaawansowana to oferuje płynniejsze rezultaty i lepiej sobie radzi z utrzymaniem szczegółów obrazu jako że wykorzystuje informacje z kilku pikseli w otoczeniu, co pozwala na bardziej zaawansowaną aproksymację wartości piksela.
- Operowanie na obrazach kolorowych jest bardziej czasochłonne