

# **Sprawozdanie z laboratorium:**

## **Informatyka w medycynie**

**Temat: Symulator tomografu komputerowego**

**Prowadzący: dr Szymon Wilk**

**Zajęcia: piątek 13:30**

**Wykonali:**

**Katarzyna Jóźwiak 127237, gr. I4**

**Piotr Pawlaczyk 127245, gr. I4**

## 1. Wstęp

Celem naszego projektu było napisanie programu symulującego działanie tomografu komputerowego.

## 2. Algorytm

Idea działania tomografu komputerowego polega na tym, że przez oryginalny obraz przepuszcza się wiązki promieniowania emitowane przez emiter do poszczególnych detektorów. Taka wiązka przechodząc przez badany obiekt zmienia swoją moc. Następnie układ emitera i detektorów jest obracany o dany kąt i czynność jest powtarzana. Zbiór pomiarów dokonanych przez detektory tworzy sinogram, za pomocą którego da się odtworzyć oryginalny obraz.

W naszej implementacji tomografu komputerowego zastosowaliśmy stożkowy układ emitera i detektorów. Na początku tworzymy tablicę współrzędnych emiterów i detektorów dla pierwszego układu. Dwa przeciwległe detektory z emitorem tworzą kąt  $\phi$  podawany przez użytkownika jako parametr programu. Następnie za pomocą algorytmu Bresenhama obliczamy średnią kolorów na kolejnych liniach emiter-detektor. Otrzymane wyniki zapisujemy w pierwszym wierszu sinogramu. Kolejnym krokiem jest obliczenie koordynatów emitera i detektorów dla układu obróconego o zadany przez użytkownika kąt  $\alpha$ . Układ obracamy o kąt  $\alpha$  aż do wykonania pełnego obrotu wokół koła.

Dzięki sinogramowi wykonanemu w ten sposób możliwe jest późniejsze odtworzenie oryginalnego obrazu. Poszczególne wiersze sinogramu są wartościami, które należy dodać na danej linii obrazu (także wyznaczanej na podstawie algorytmu Bresenhama) przeprowadzonej między detektorem a emitorem. Taka czynność jest powtarzana tyle razy, ile razy układ został odwrócony do uzyskania 360 stopni.

Aby ulepszyć jakość uzyskanego w ten sposób obrazu zastosowaliśmy filtr Ram-Lak. Filtr ten jest symetryczny określony wzorem:

EQUATION 25-2  
The filter kernel for filtered  
backprojection. Figure 25-19b  
shows a graph of this kernel.

$$h[0] = 1$$

$$h[k] = 0 \quad \text{for even values of } k$$

$$h[k] = \frac{-4/\pi^2}{k^2} \quad \text{for odd values of } k$$

W naszej implementacji programu filtr ma długość równą  $1/20$  szerokości sinogramu.

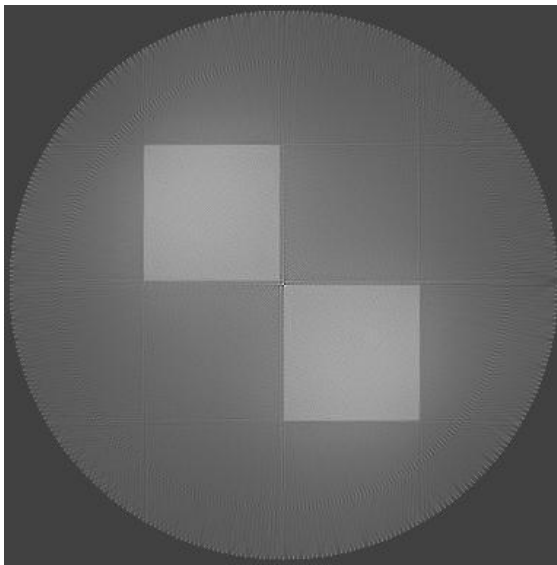
Filtrowanie polega na splocie kolejnych wierszy oryginalnego sinogramu z wcześniej stworzonym filtrem. Następnie z tak przefiltrowanego sinogramu odzyskiwany jest oryginalny wejściowy obraz. Zastosowanie filtru sprawia, że jest on dużo wyraźniejszy.

### 3. Testowanie

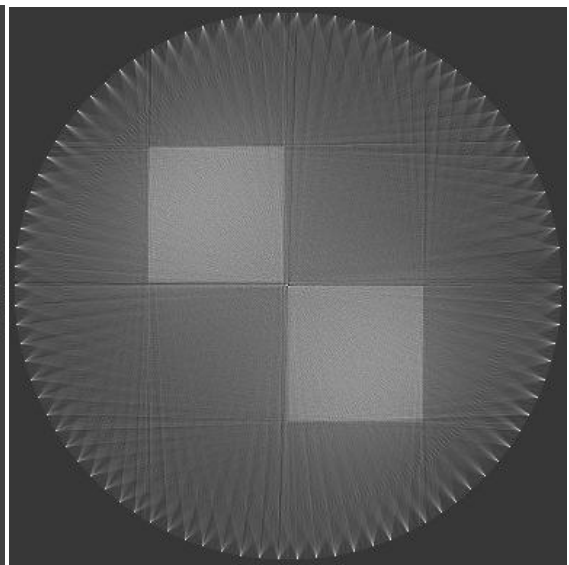
Parametrami podawanymi przez użytkownika na wejście programu jest ilość detektorów, kąt rozwarcia detektorów  $\phi$ , kąt obrotu układu w kolejnych iteracjach oraz to czy obraz ma zostać przefiltrowany czy nie.

#### a) Zmienny kąt obrotu układu alfa

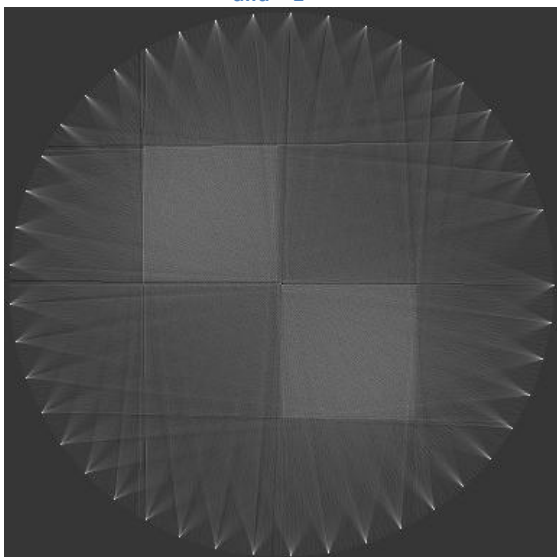
Pierwszym testowanym przez nas parametrem był kąt alfa. Badaliśmy jaki ma on wpływ na jakość uzyskanego obrazu. Badaliśmy jakość obrazu poprzez generowanie plików o stałych wartościach kąta  $\phi$  (250 stopni) i liczby detektorów (281). Dla tak generowanych obrazów zmienialiśmy tylko wartość kąta alfa. W poniższych obrazkach zastosowaliśmy filtr.



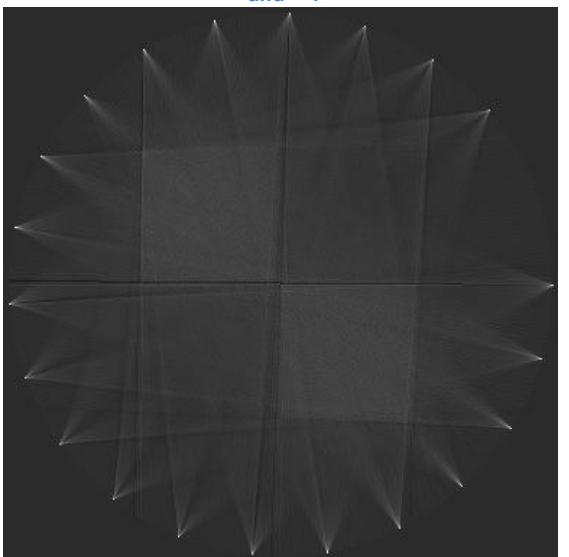
alfa = 1



alfa = 4



alfa = 8



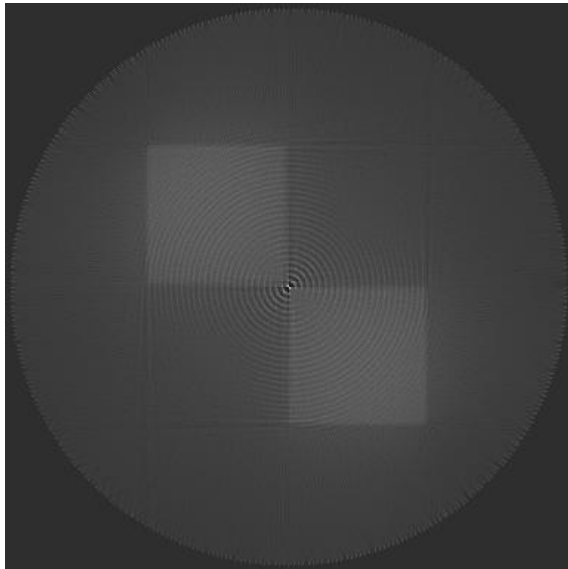
alfa = 16

Zauważyliśmy, że tym większy kąt obrotu układu, tym bardziej niejednolite jest tło obrazu. Ponadto obraz staje się coraz bardziej rozmyty. Jest to wynikiem tego, że zwiększając kąt obrotu,

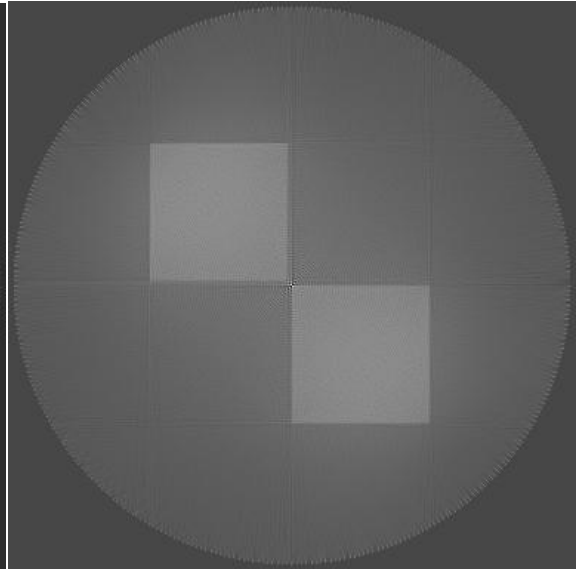
automatycznie sprawiamy, że potrzeba mniej obrotów układu emiter – detektory do uzyskania pełnego obrotu, a o za tym idzie, dokonywanych jest mniej pomiarów.

### **b) Zmienna liczba detektorów**

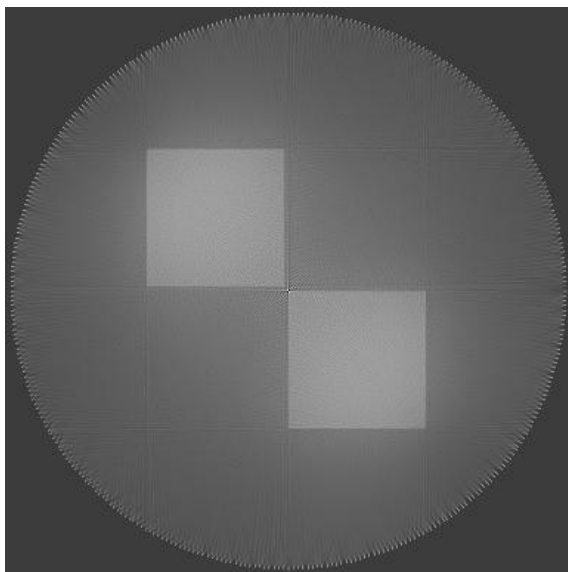
Kolejnym badanym elementem był wpływ liczby detektorów na jakość obrazu. Ten test przeprowadziliśmy dla kąta alfa równego 1 a kąta fi równego 250.



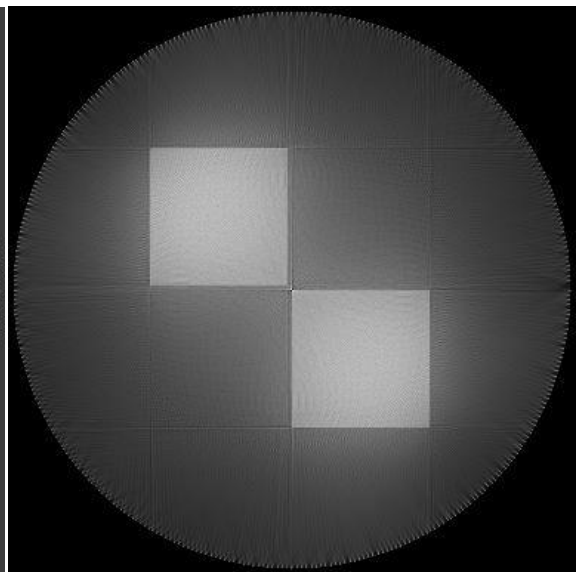
liczba detektorów = 101



liczba detektorów = 201



liczba detektorów = 251

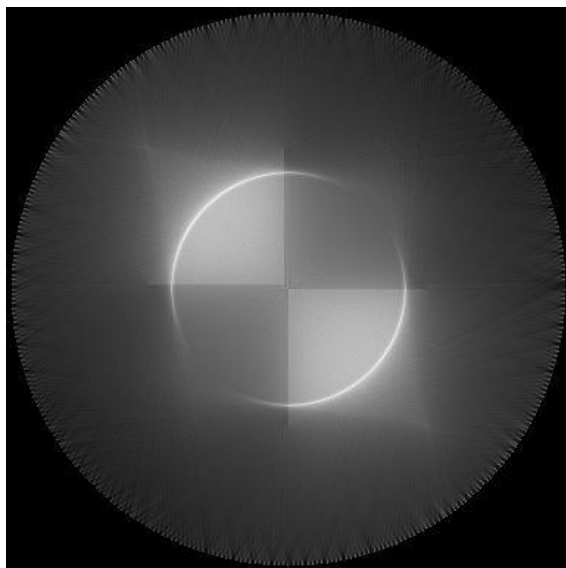


liczba detektorów = 351

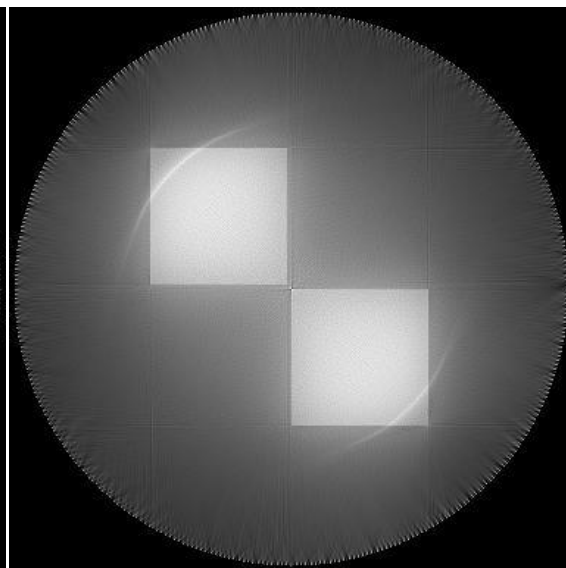
Z powyższych zdjęć wynika, że im większa liczba detektorów tym lepsza jakość obrazu (lepszy kontrast kolorów).

### c) Zmienny kąt rozwarcia detektorów $\phi$

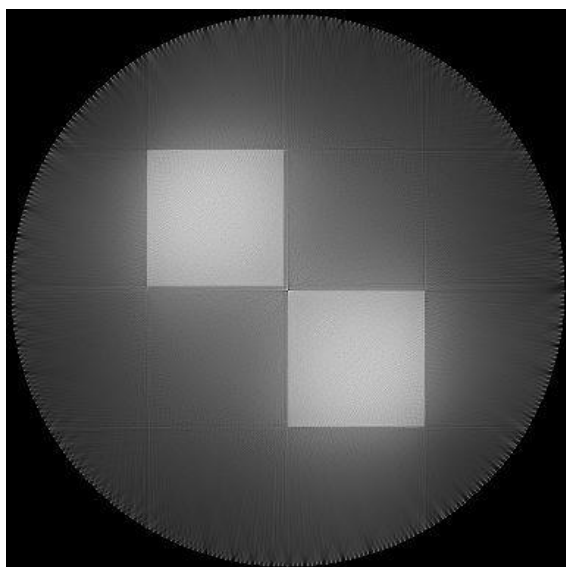
Ostatnim badanym parametrem był kąt rozwarcia detektorów  $\phi$ . Badaliśmy go dla kąta  $\alpha = 1$  stopień i liczby detektorów równej 301.



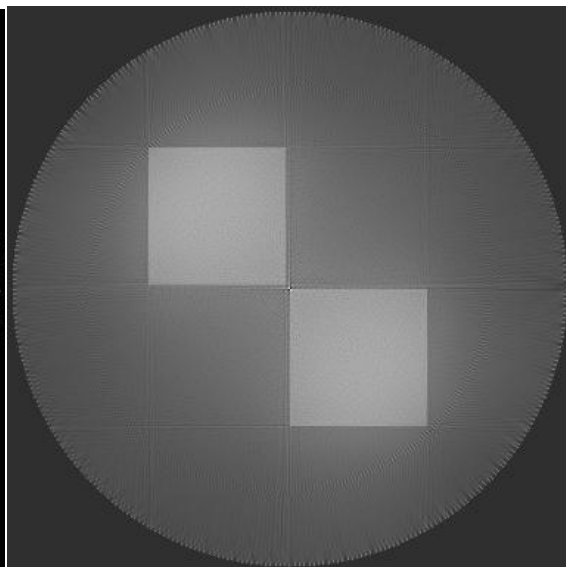
kąt  $\phi = 100$



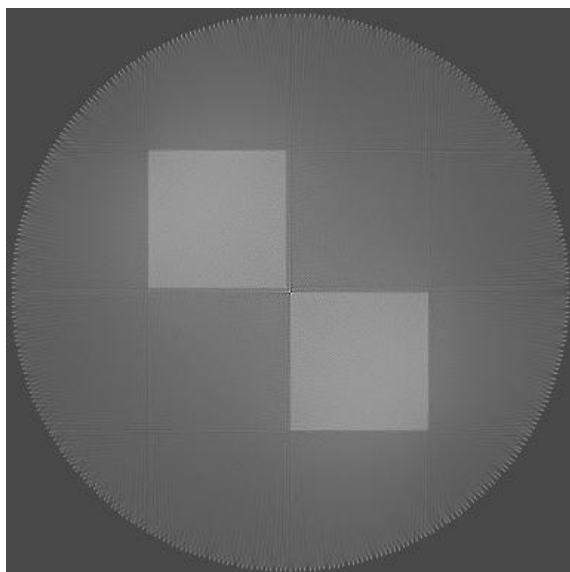
kąt  $\phi = 150$



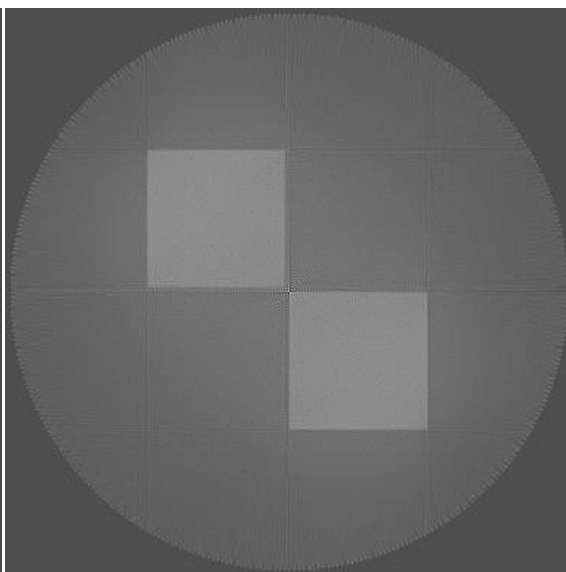
kąt  $\phi = 200$



kąt  $\phi = 250$



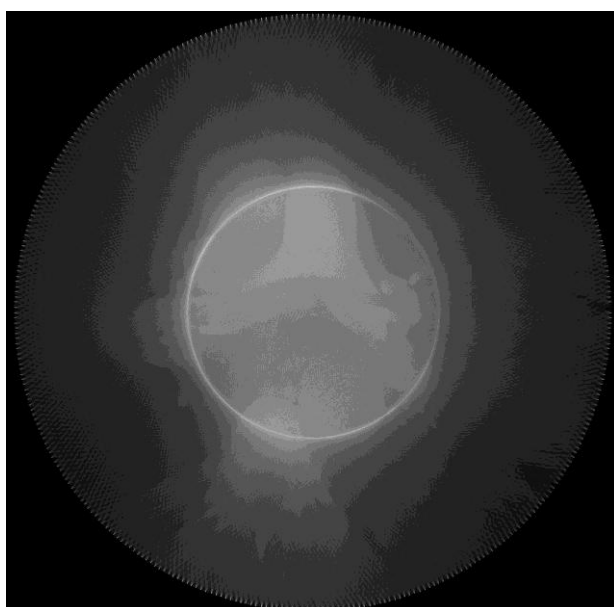
kąt  $\phi$  = 300



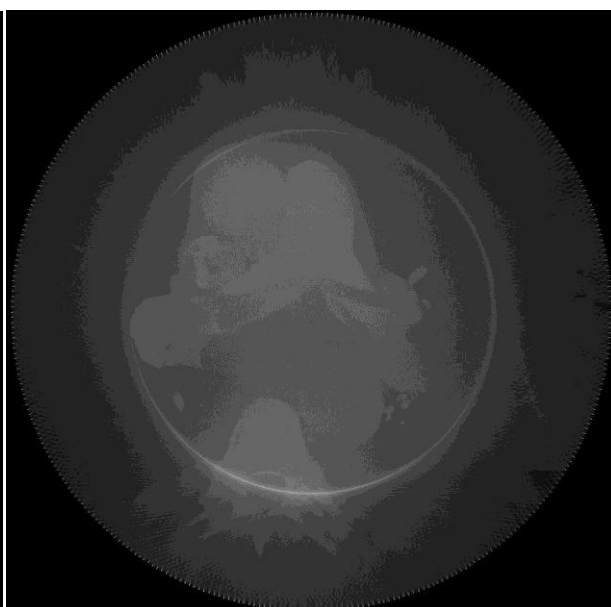
kąt  $\phi$  = 350

Zmiana kąta rozwarcia detektorów  $\phi$  ma bardzo duży wpływ na jakość obrazu. Dla  $\phi=100$  stopni brzegi obrazu są rozmyte i widać wyraźną granicę między prawidłowym i ostrym a rozmytym obrazem. Po osiągnięciu pewnej wartości kąta  $\phi$ , ta różnica przestaje być widoczna.

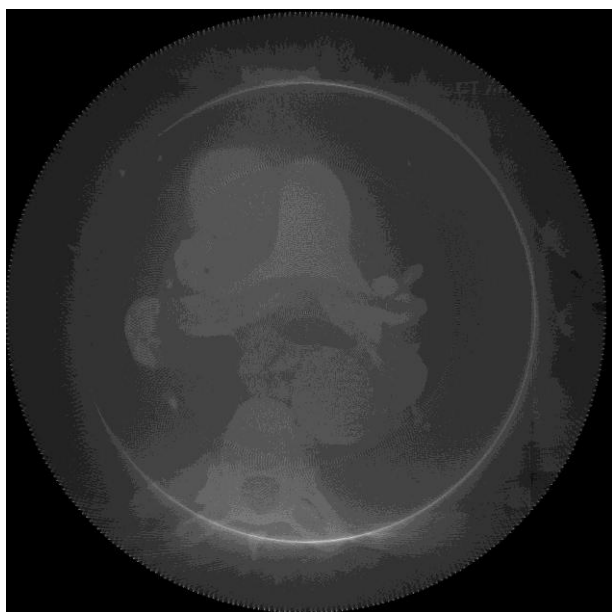
Takie zjawisko jest też bardzo wyraźne w przypadku innych, bardziej klinicznych obrazów jak np. SADDLE\_PE. W tym przypadku widać, że wstawiając za parametr zbyt duży kąt rozwarcia  $\phi$ , efekt jest taki, że obraz zaczyna mieć zbyt mały kontrast.



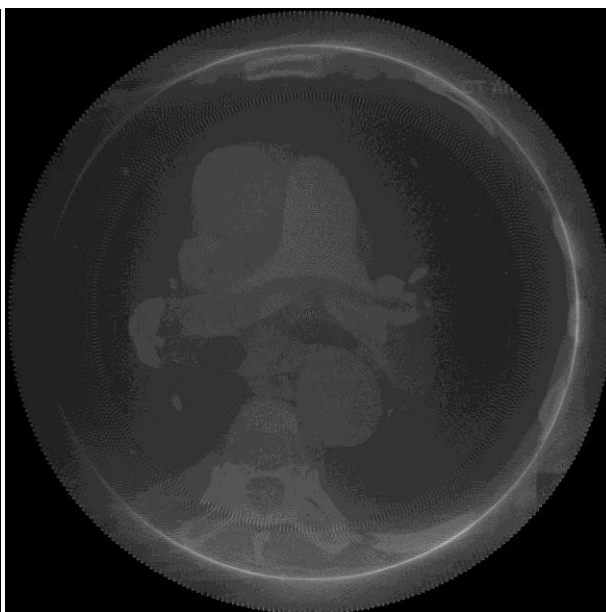
kąt  $\phi$  = 150



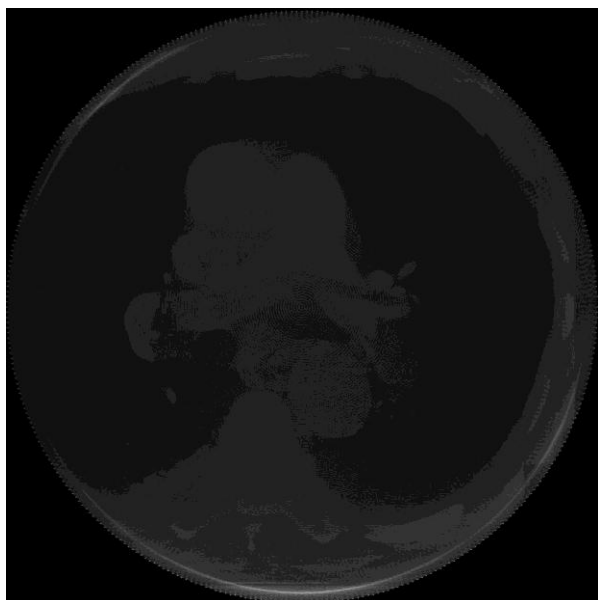
kąt  $\phi$  = 100



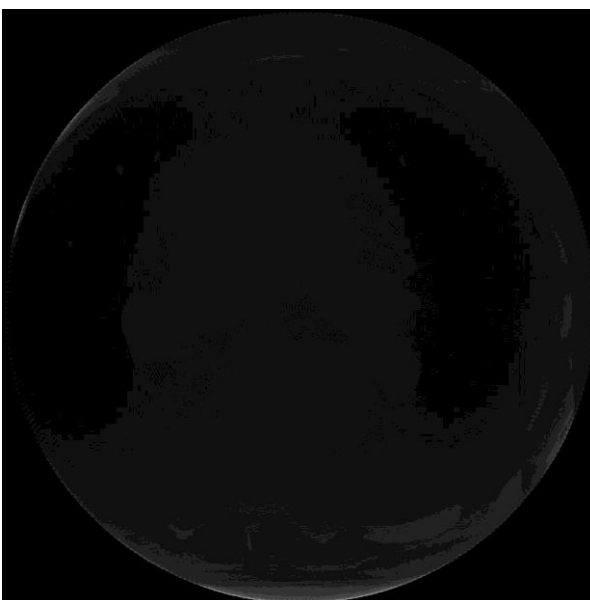
kąt  $\phi_i = 250$



kąt  $\phi_i = 200$



kąt  $\phi_i = 350$

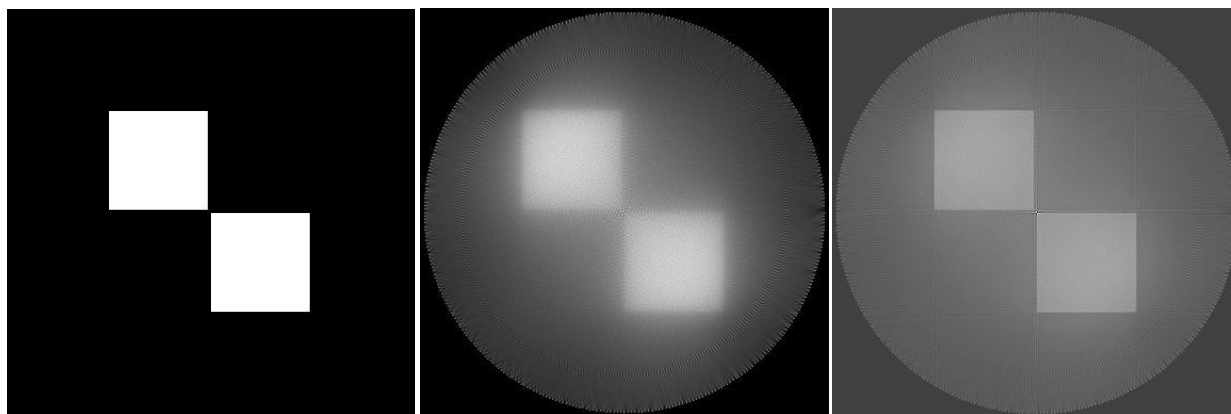


kąt  $\phi_i = 300$

#### **d) Badanie zmiany poszczególnych parametrów na błąd średniokwadratowy**

Przedmiotem kolejnego testu było badanie zmiany poszczególnych parametrów programu na błąd średniokwadratowy. Trudno było przeprowadzić badanie błędu średniokwadratowego dla porównania obrazu bez zastosowania filtru i z filtrem w stosunku do oryginalnego z uwagi na to, że błąd średniokwadratowy badaliśmy na podstawie zmiany koloru, a obraz po filtracji, mimo wyraźnej poprawy jakości i ostrości obrazu, miał jaśniejsze tło (które w oryginalnym obrazie było czarne), co sprawia, że wynik jest niemiernie dokładny.





Oryginalny obraz

Obraz wyn. bez zastosowania filtru

Obraz wyn. z zastosowaniem filtru

Dla przykładu zamieścimy zdjęcie kwadratów. Wyraźnie widać, że zastosowanie filtru wyostrza obraz, jednak obraz bez zastosowania filtru mierzony miarą błędów średniokwadratowych będzie bardziej zbliżony do oryginalnego obrazu z uwagi na wyraźniejszą kolorystykę

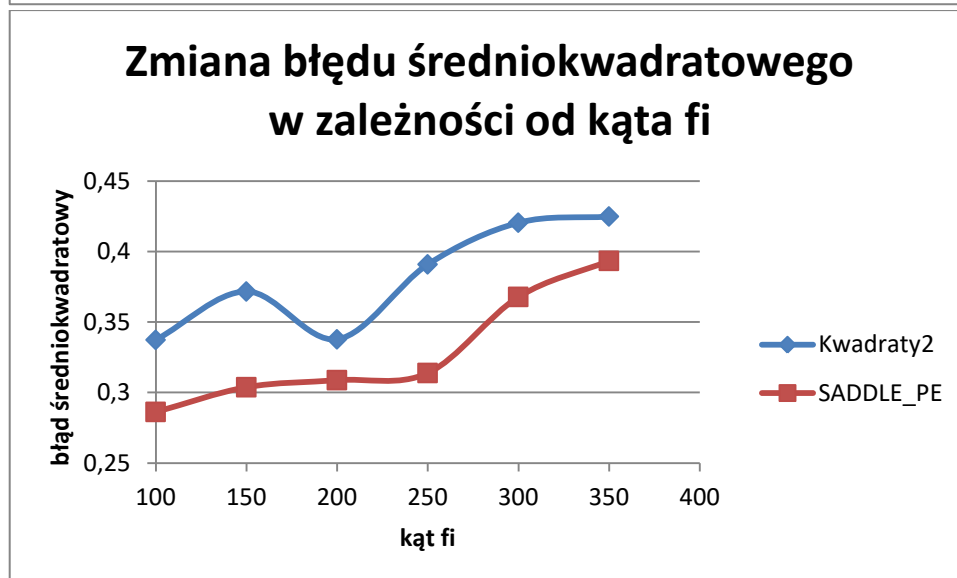
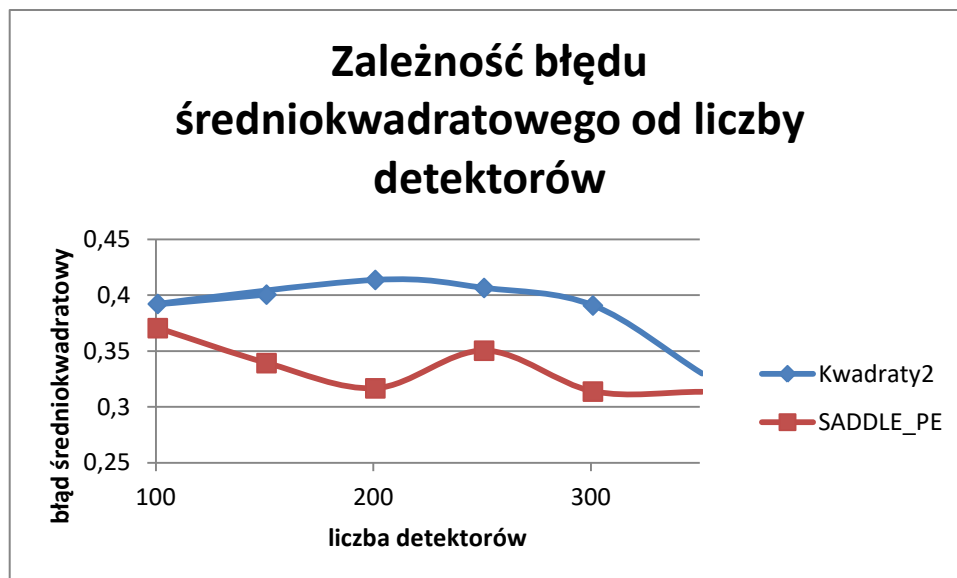
	Błąd średniokwadratowy
Bez zastosowania filtru	0,36229474484
Z zastosowaniem filtru	0,41520881672

W kolejnych testach zbadaliśmy wpływ zmiany parametrów w poszczególnych obrazach (z zastosowaniem filtru) na błąd średniokwadratowy.

Do liczenia błędów średniokwadratowych wykorzystaliśmy obszar znajdujący się w środku okręgu tworzonego przez detektory.





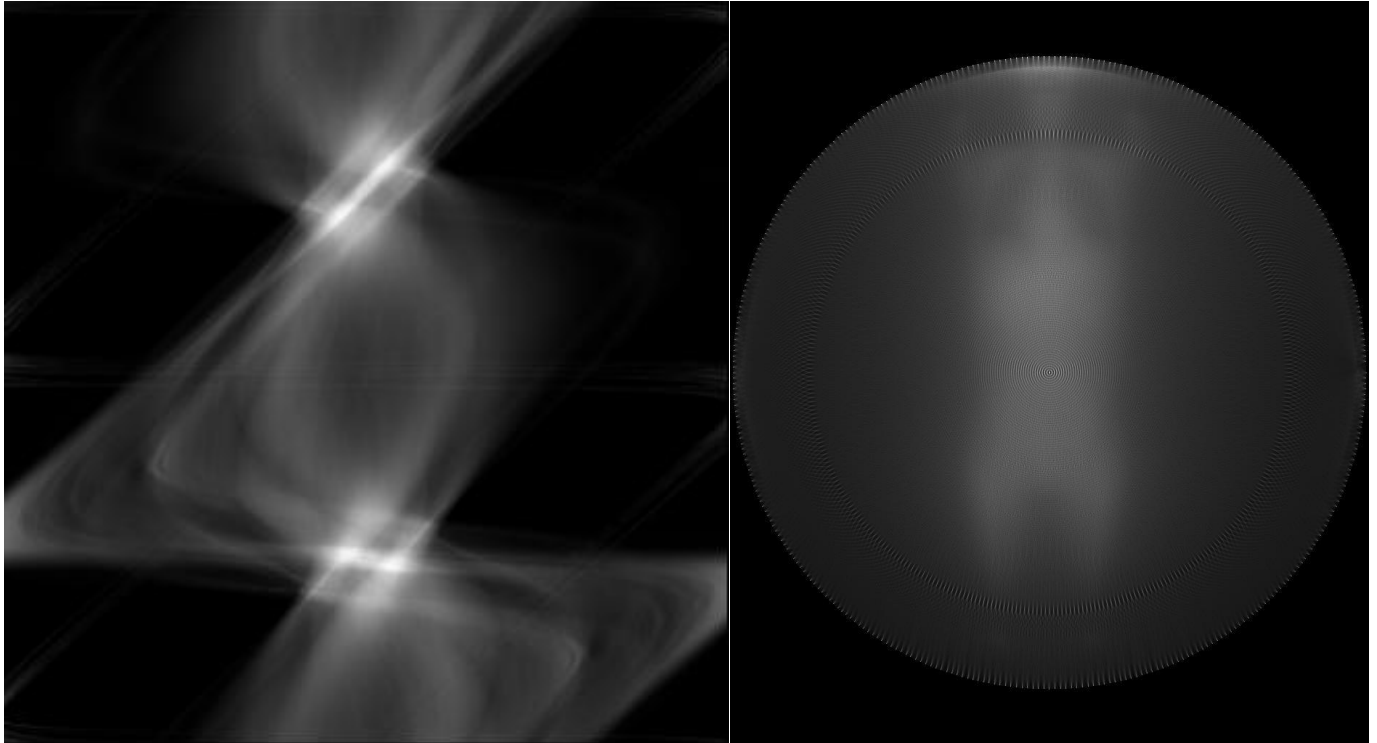


## 5. Przykłady innych obrazów i ich sino gramów

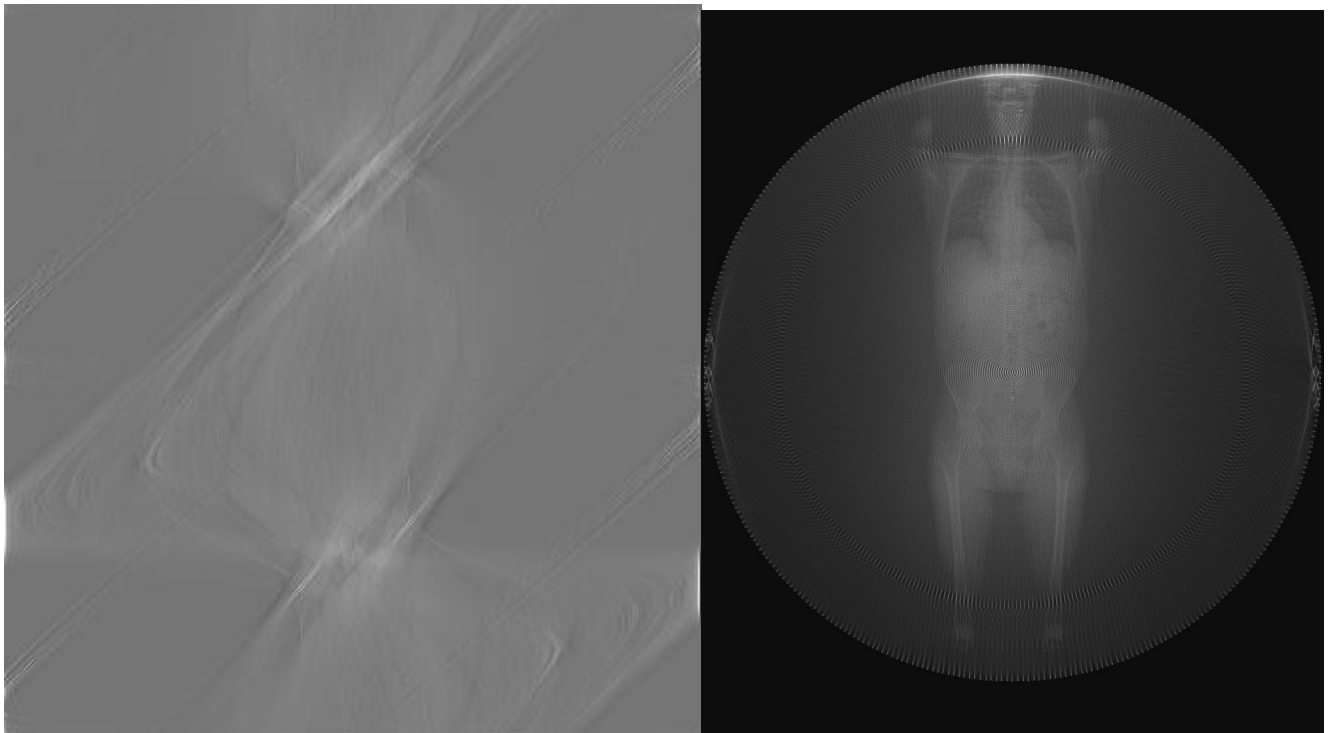
W poniższym zestawieniu użyliśmy tych parametrów, które w poszczególnych testach wydawały się dawać najlepsze wyniki, czyli:

- kąt  $\alpha$  = 1 stopień,
- liczba detektorów = 351
- kąt  $\phi$  = 300 stopni

a) CT\_ScoutView

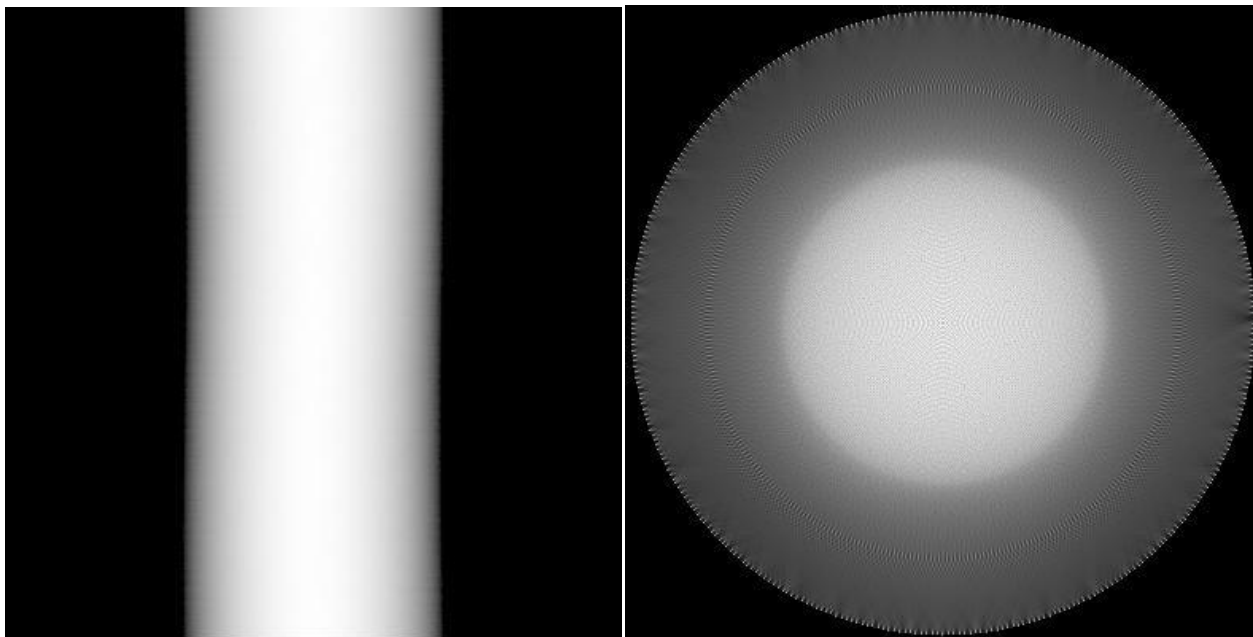


sinogram i obraz wynikowy bez zastosowania filtra

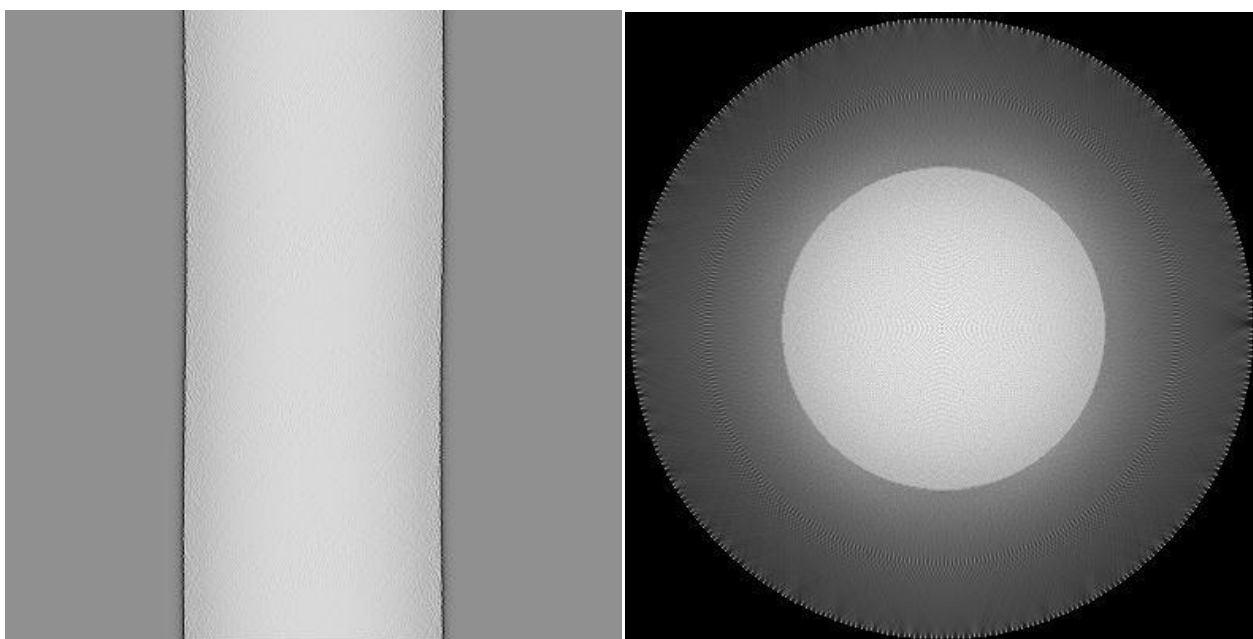


sinogram i obraz wynikowy z zastosowaniem filtra

**b) Koło**

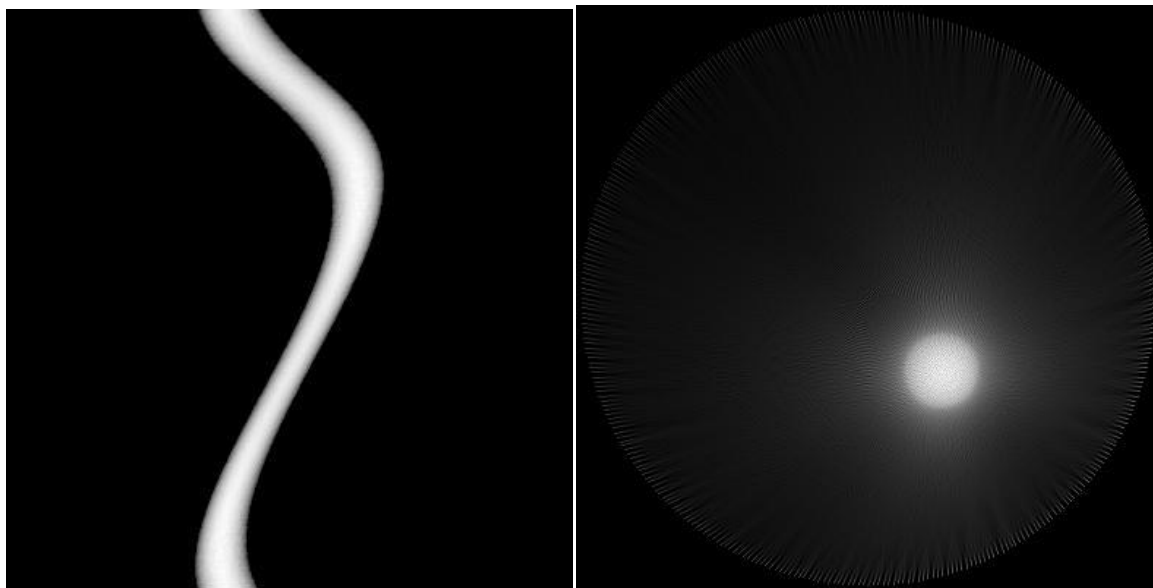


sinogram i obraz wynikowy bez zastosowania filtra

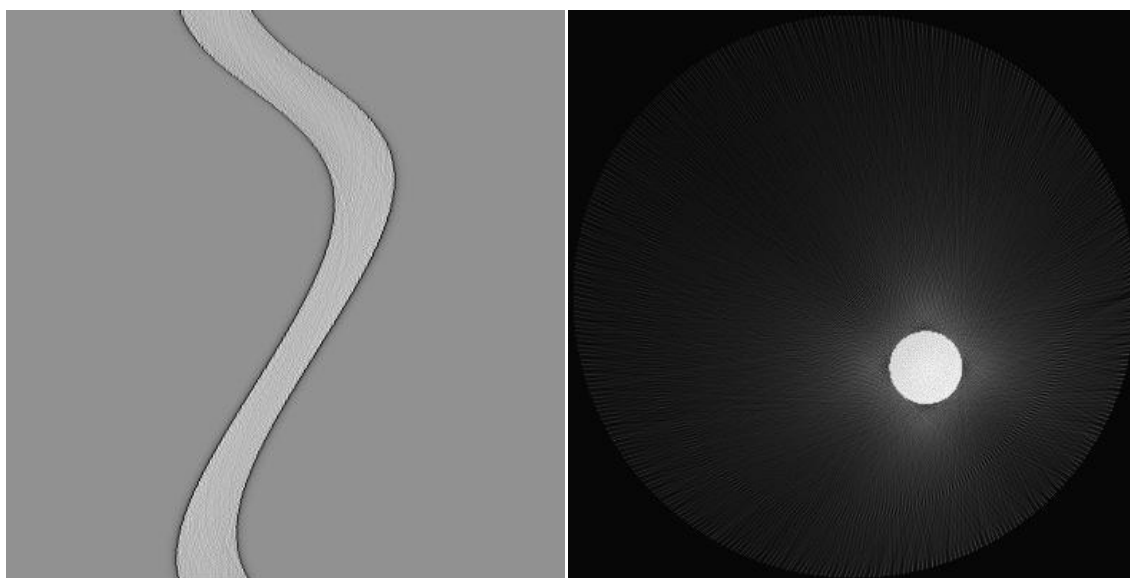


sinogram i obraz wynikowy z zastosowaniem filtra

c) Kropka

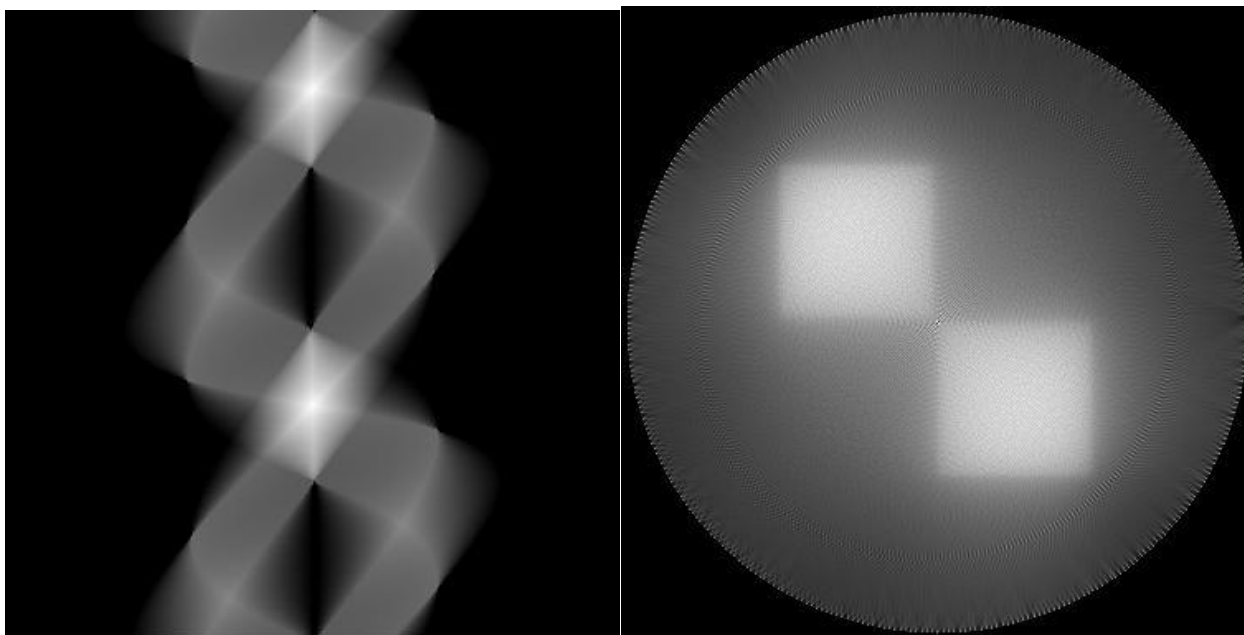


sinogram i obraz wynikowy bez zastosowania filtra

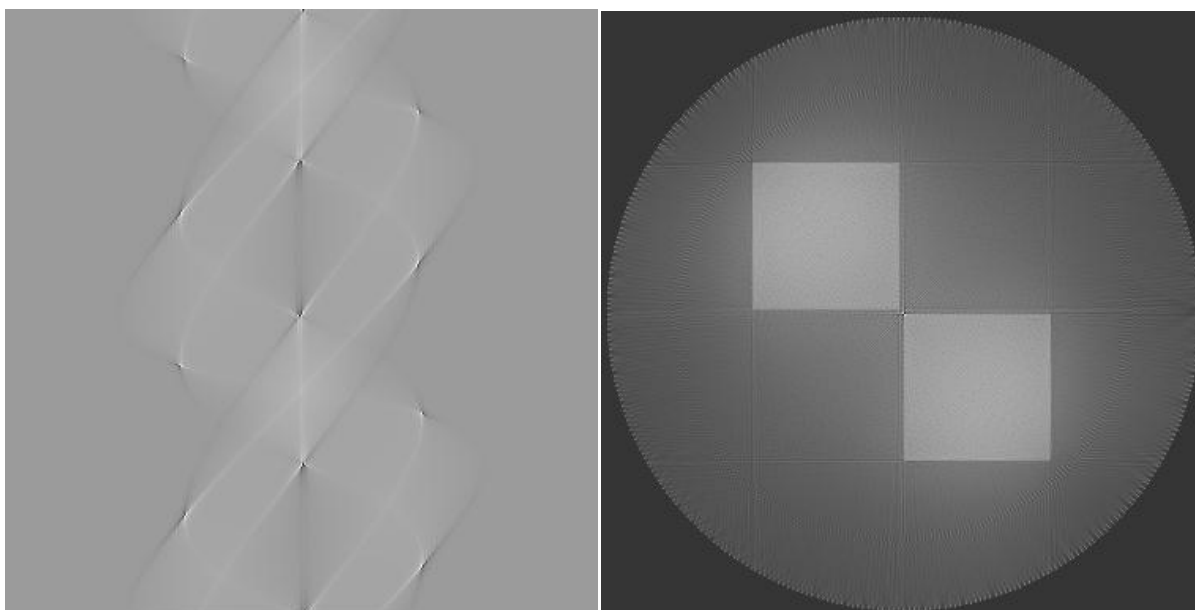


sinogram i obraz wynikowy z zastosowaniem filtra

#### d) Kwadraty2

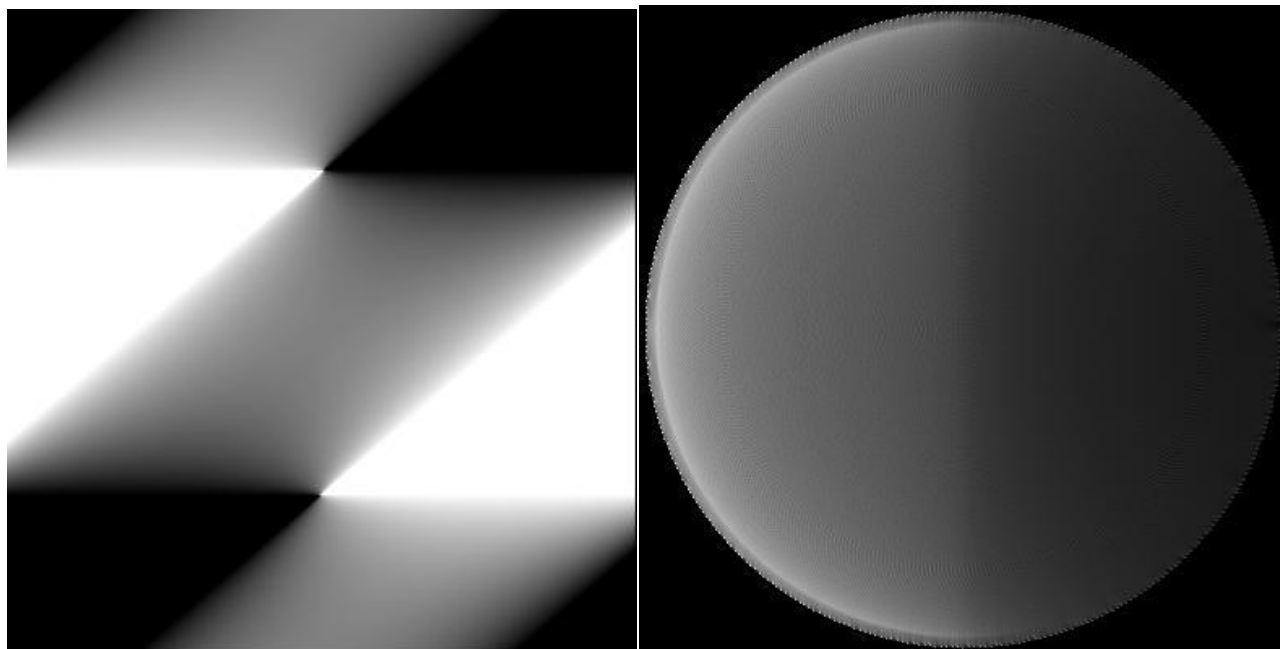


sinogram i obraz wynikowy bez zastosowania filtra

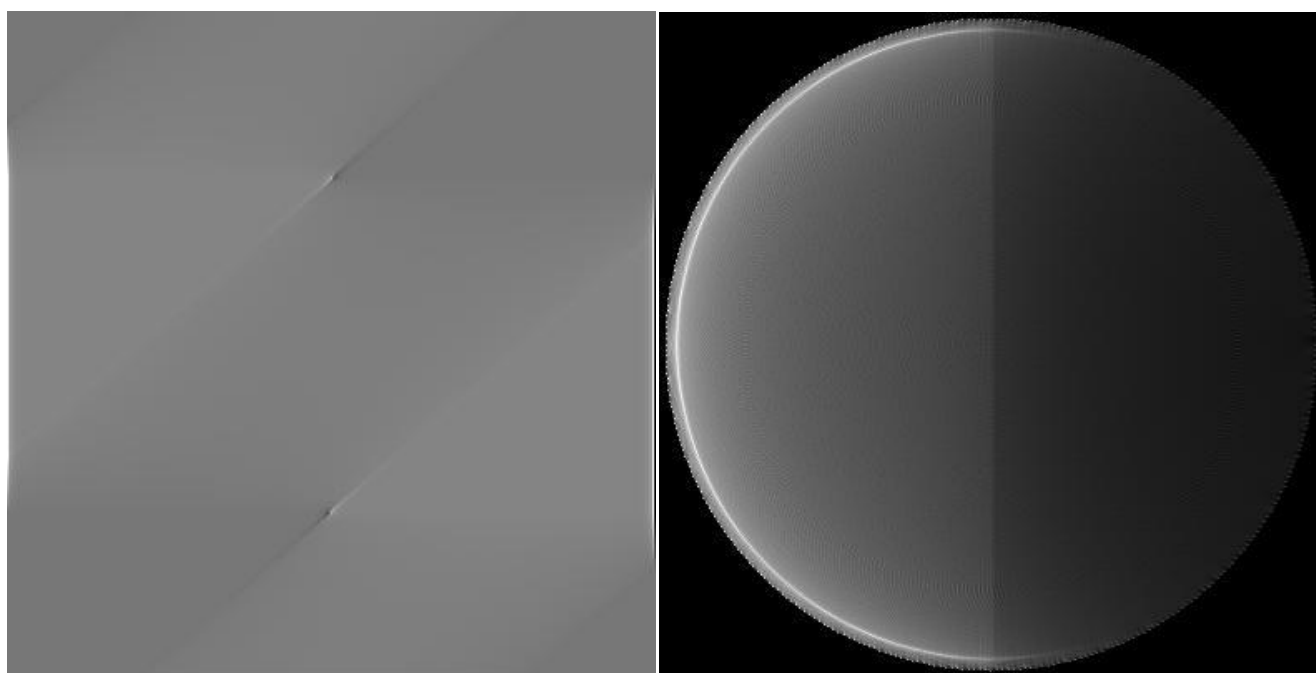


sinogram i obraz wynikowy z zastosowaniem filtra

**e) Paski2**

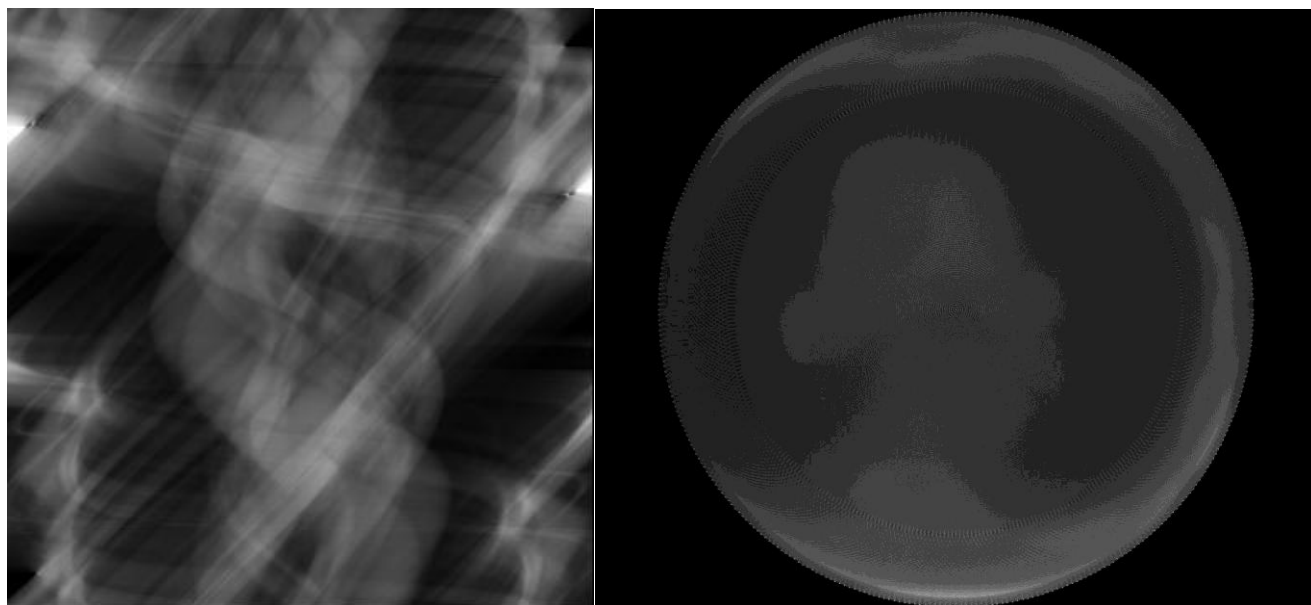


sinogram i obraz wynikowy bez zastosowania filtra

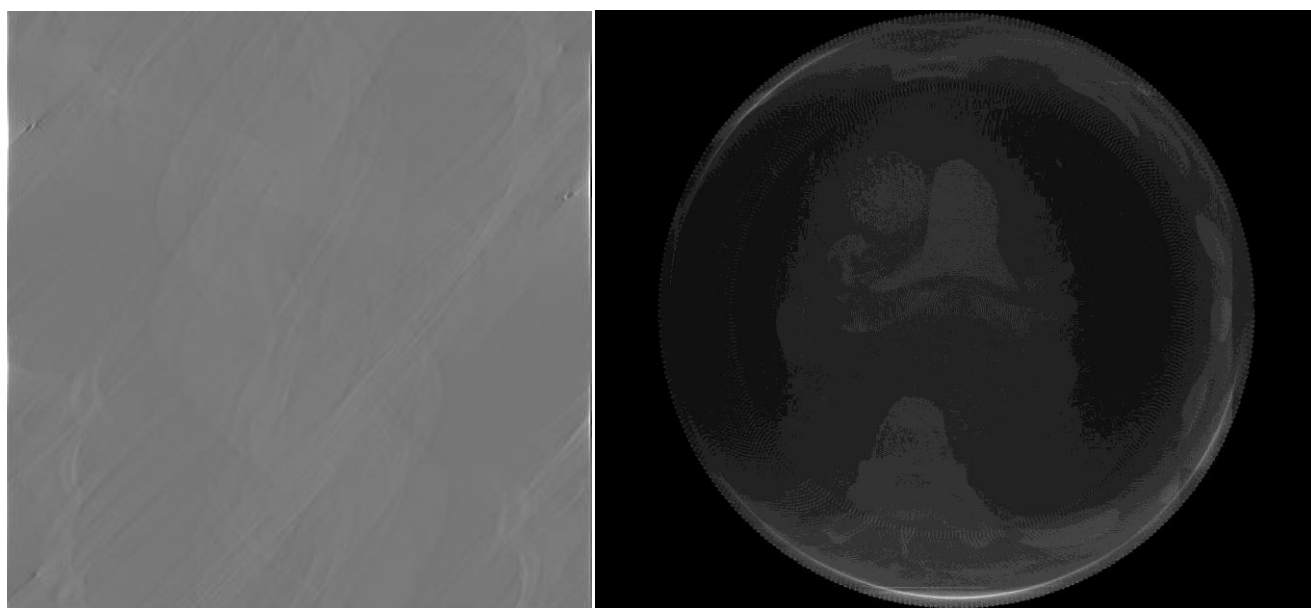


sinogram i obraz wynikowy z zastosowaniem filtra

f) SADDLE\_PE



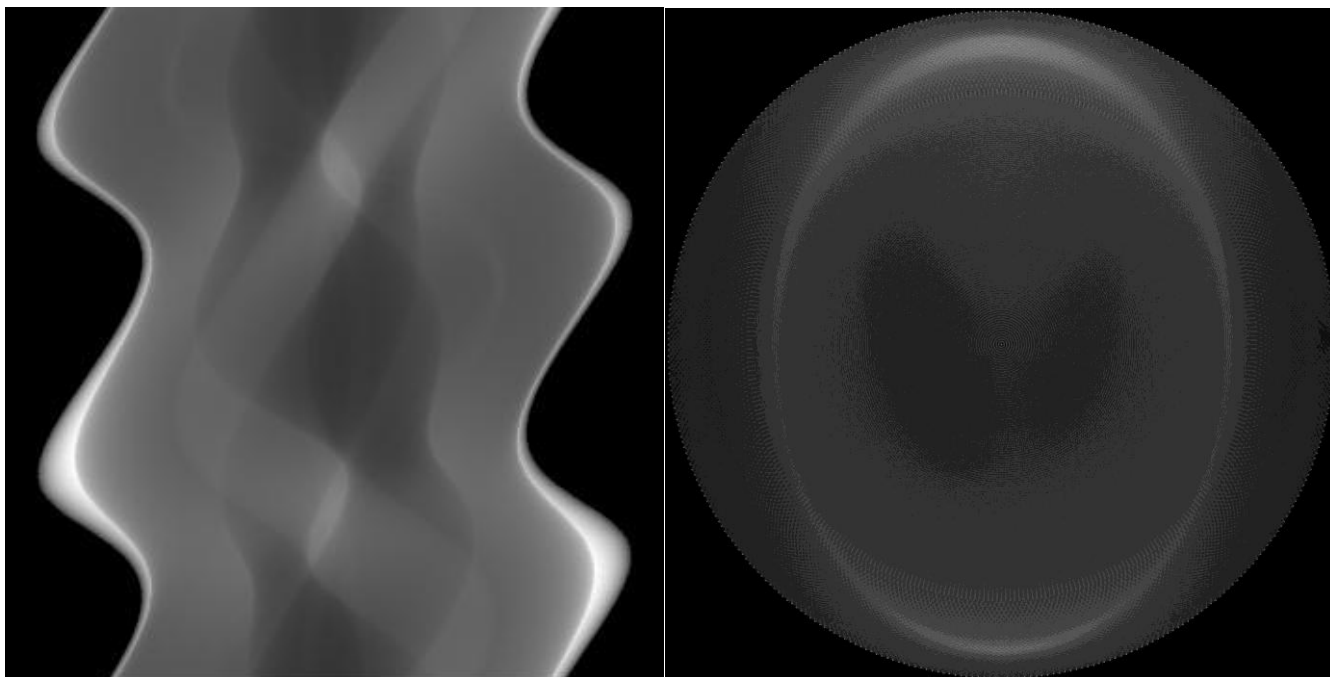
sinogram i obraz wynikowy bez zastosowania filtra



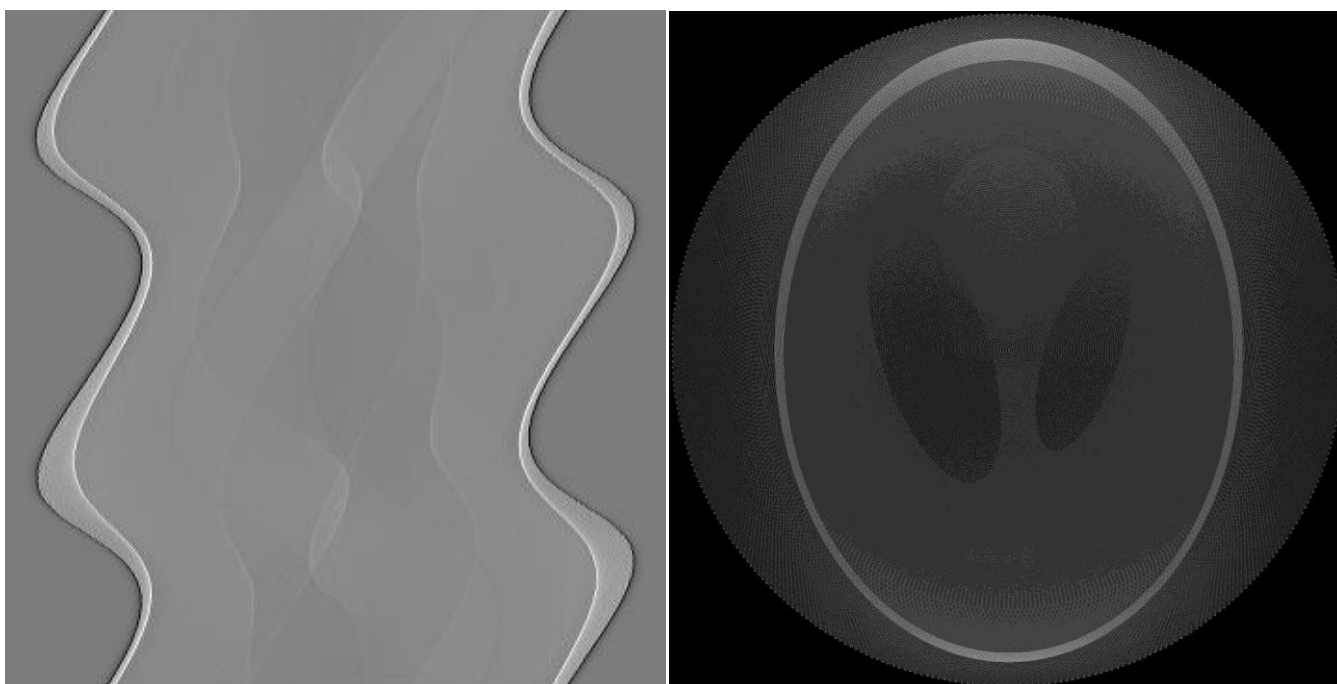
sinogram i obraz wynikowy z zastosowaniem filtra



g) Sheep\_logan



sinogram i obraz wynikowy bez zastosowania filtra



sinogram i obraz wynikowy z zastosowaniem filtra