

Potokowa realizacja trójwymiarowego filtra uśredniającego w oparciu o struktury rotatorowe

Streszczenie. Artykuł dotyczy systemu przetwarzania danych 3D. Zaprezentowana została synteza potokowej realizacji filtra uśredniającego 3D z wykorzystaniem struktur rotatorowych. W procesie syntezy zastosowano oryginalne techniki pozwalające na uzyskanie układu o porównywalnych lub lepszych parametrach w porównaniu do struktur bezpośrednich. System zweryfikowany został symulacyjnie w środowisku Scilab, przetworzone zostały również przykładowe obrazy 3D. Podczas realizacji sprzętowej, rotatory pozwalają na zminimalizowanie złożoności obliczeniowej struktury oraz zmniejszenie wrażliwości układu na skończoną precyzję obliczeń. Zastosowanie algorytmu CORDIC przy implementacji rotatorów, umożliwiło uzyskanie w pełni potokowego systemu.

Abstract. This paper presents synthesis of a pipeline three-dimensional (3D) mean filter by means of rotation structure. The synthesis is based on original method for separable FIR filters. Rotation structures allow to obtain smaller sensitivity of finite precision calculations and make possible to use pipelining. The system has been verified by a simulation in Scilab environment using a 3D medical image in DICOM format. (**Pipeline realization of three-dimensional mean filter using rotation structure**).

Słowa kluczowe: CPS 3D, filtr uśredniający 3D, filtr potokowy, rotator.

Keywords: 3D DSP, 3D mean filter, pipeline filter, rotation.

Wstęp

Systemy Cyfrowego Przetwarzania Sygnałów (CPS) w chwili obecnej niemal całkowicie wyparty już układy analogowe. Gwałtowny rozwój mikroprocesorów oraz metod przetwarzania danych pozwoliły na implementację coraz bardziej złożonych systemów. Pojawianie się nowych technik CPS związane jest nie tylko z doskonaleniem przeprowadzanych obliczeń, ale również z przetwarzaniem coraz większej ilości danych, co sprawia, że układy wymagają dużo większej mocy obliczeniowej. Obecnie przetwarza się głównie sygnały jedno-, dwu- oraz trójwymiarowe (1D, 2D, 3D) [1, 2], a coraz częściej również bardziej złożone struktury danych [3, 4]. Związane jest to przede wszystkim z dziedzinami, w których najczęściej spotyka się układy CPS 3D, np. medycyna [5], analiza materiałów [6], badania gruntów [7], itp. Duża złożoność nawet podstawowych systemów filtracji 3D, które najczęściej oparte są na bezpośredniej operacji splotu (konieczna duża precyzja obliczeń), sprawiła że wymagają one znacznych zasobów sprzętowych. Wszystko to ma wpływ na potrzebę rozwoju technik syntezy systemów CPS, tak aby implementowane układy nie odbiegały parametrami od bezpośrednich metod (splot), a jednocześnie ilość wykorzystywanych złożonych obliczeń była minimalna, co ma bezpośrednie przełożenie na zmniejszenie wymagań sprzętowych.

Prezentowany artykuł dotyczy syntezy potokowego filtra uśredniającego 3D z zastosowaniem struktur rotatorowych i permutacji [8, 9, 10], co docelowo powinno pozwolić na istotną redukcję złożoności obliczeniowej całego układu i przy odpowiednich technikach implementacji sprzętowej, poprawę wybranych parametrów [11]. Rotator Givensa, który ma być podstawowym elementem składowym finalnego systemu, można opisać następująco:

$$(1) \quad \begin{cases} y_s = u_s \cos(\alpha) - u_t \sin(\alpha) \\ y_t = u_s \sin(\alpha) + u_t \cos(\alpha) \end{cases},$$

natomiast permutacje przedstawić można w następujący sposób:

$$(2) \quad \begin{cases} y_s = u_t \\ y_t = u_s \end{cases},$$

gdzie: s, t – indeksy wejść i wyjść, których wartości zostaną przekształcone.

Punktem początkowym syntezy jest maska współczynników, która dla filtra uśredniającego o wymiarach $3 \times 3 \times 3$ [12], wygląda następująco:

$$(3) \quad H = \begin{bmatrix} 1/27 & 1/27 & 1/27 \\ 1/27 & 1/27 & 1/27 \\ 1/27 & 1/27 & 1/27 \end{bmatrix} \begin{bmatrix} 1/27 & 1/27 & 1/27 \\ 1/27 & 1/27 & 1/27 \\ 1/27 & 1/27 & 1/27 \end{bmatrix} \begin{bmatrix} 1/27 & 1/27 & 1/27 \\ 1/27 & 1/27 & 1/27 \\ 1/27 & 1/27 & 1/27 \end{bmatrix}.$$

Poniżej przedstawione są kolejne etapy syntezy, które zaimplementowane zostały w środowisku Scilab [13], umożliwiającym dokonywanie złożonych działań i operacji macierzowych.

Synteza potokowej realizacji filtra

Maska (3) może być zapisana w postaci transmitancji 3D:

$$(4) \quad H(z_h, z_v, z_d) = H_O(z_h, z_v)z_d^{-2} + H_O(z_h, z_v)z_d^{-1} + H_O(z_h, z_v),$$

gdzie:

$$H_O(z_h, z_v) = \frac{1}{27}z_h^{-2}z_v^{-2} + \frac{1}{27}z_h^{-1}z_v^{-2} + \frac{1}{27}z_v^{-2} + \frac{1}{27}z_h^{-2}z_v^{-1} + \frac{1}{27}z_h^{-1}z_v^{-1} + \frac{1}{27}z_v^{-1} + \frac{1}{27}z_h^{-2} + \frac{1}{27}z_h^{-1} + \frac{1}{27}$$

Transmitancje (4) dla filtrów separowalnych [10], daje się zapisać w postaci iloczynu trzech transmitancji jednowymiarowych:

$$(5) \quad H(z_h, z_v, z_d) = H_h(z_h)H_v(z_v)H_d(z_d).$$

gdzie:

$$(6) \quad H_h(z) = H_v(z) = H_d(z) = \frac{1}{3}z^{-2} + \frac{1}{3}z^{-1} + \frac{1}{3}.$$

Zgodnie z [8], transmitancje 1D (6) muszą spełniać warunek $|H(e^{j\omega})| \leq 1$, w wyniku czego otrzymujemy dodatkowe współczynniki ($k_h = k_v = k_d = 1$), które ze względu na specyfikę filtra uśredniającego mogą być pominięte (rys. 1). Transmitancje (6) mogą być przekształcone do postaci trzech układów równań stanu [8, 10]:

$$(7) \quad H_h(z_h) \Rightarrow S_h = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad H_v(z_v) \Rightarrow S_v = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad H_d(z_d) \Rightarrow S_d = \begin{bmatrix} A & B \\ C & D \end{bmatrix},$$

gdzie:

$$(8) \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} 0,4058277 & -0,4526419 & -0,6886331 \\ 0,3638552 & -0,4058277 & 0 \\ -0,4840507 & -0,7904504 & 0,3333333 \\ 0,6845496 & -0,0748826 & 0,6439513 \end{bmatrix}$$

Następnie, (8) należy doprowadzić do postaci macierzy kwadratowej z zachowaniem warunku ortogonalności [9], poprzez dodanie dodatkowego wejścia:

$$(9) \begin{bmatrix} A_o & B_o \\ C_o & D_o \end{bmatrix} = \begin{bmatrix} 0,4058277 & -0,4526419 & -0,6886331 & -0,3952260 \\ 0,3638552 & -0,4058277 & 0 & 0,8383992 \\ -0,4840507 & -0,7904504 & 0,3333333 & -0,1725458 \\ 0,6845496 & -0,0748826 & 0,6439513 & -0,3333333 \end{bmatrix}$$

W celu zmniejszenia liczby finalnych rotatorów i permutacji, konieczne jest zminimalizowanie niezerowych elementów w macierzach A_o subsytemów 1D (6) tworząc macierze trójkątne górne [9]:

$$(10) \begin{bmatrix} A_h & B_h \\ C_h & D_h \end{bmatrix} = \begin{bmatrix} A_v & B_v \\ C_v & D_v \end{bmatrix} = \begin{bmatrix} A_d & B_d \\ C_d & D_d \end{bmatrix} = \begin{bmatrix} 0 & 0,8164971 & 0,5127291 & -0,2654078 \\ 0 & 0 & 0,4597004 & 0,8880741 \\ 0,8880741 & -0,2654078 & 0,3333333 & -0,1725458 \\ -0,4597004 & -0,5127291 & 0,6439513 & -0,3333333 \end{bmatrix}$$

Cały system (4) opisany równaniami stanu (10) można przedstawić w następujący sposób:

$$(11) \begin{aligned} x_h(h+1, v, d) &= A_h \cdot x_h(h, v, d) + B_h \cdot u(h, v, d) \\ y_h(h, v, d) &= C_h \cdot x_h(h, v, d) + D_h \cdot u(h, v, d) \\ x_v(h, v+1, d) &= A_v \cdot x_v(h, v, d) + B_v \cdot y_h(h, v, d) \\ y_v(h, v, d) &= C_v \cdot x_v(h, v, d) + D_v \cdot y_h(h, v, d) \\ x_d(h, v, d+1) &= A_d \cdot x_d(h, v, d) + B_d \cdot y_v(h, v, d) \\ y_d(h, v, d) &= C_d \cdot x_d(h, v, d) + D_d \cdot y_v(h, v, d) \end{aligned}$$

gdzie: $u(h, v, d)$ - wejście systemu, $y(h, v, d)$ - wyjście systemu, $y_h(h, v, d)$, $y_v(h, v, d)$ - sygnały przejściowe subsytemów, $x_h(h, v, d)$, $x_v(h, v, d)$, $x_d(h, v, d)$ - wektory stanów subsytemów.

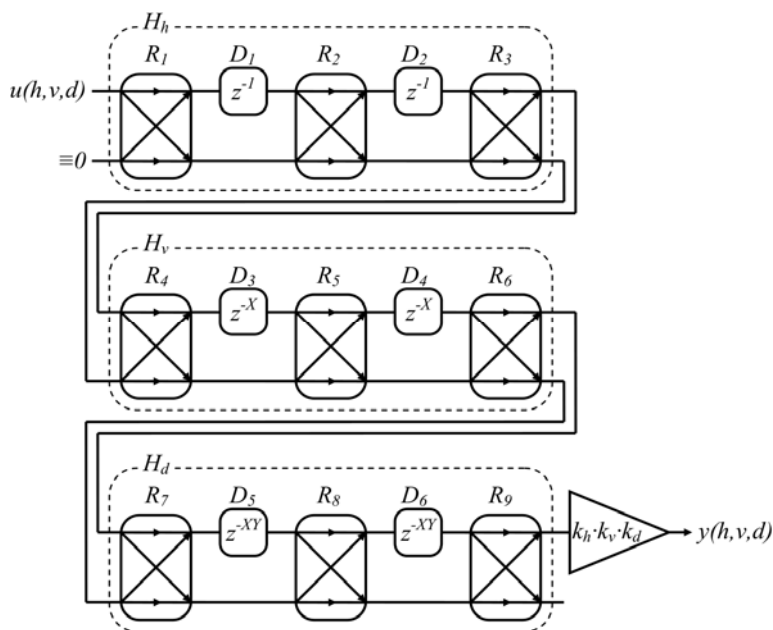
Tak przygotowane macierze równań stanu (10), w celu otrzymania struktury rotatorowej, poddawane są działaniu odpowiedniego algorytmu QR [9]. Ostatecznie każdy subsystem (6), z postaci macierzy równań stanu (10) jest przekształcony do iloczynu wielu macierzy, które są odpowiednimi macierzami rotacji i permutacji.

Tabela 1. Parametry rotatorów i permutacji

R_i/P_i		$\sin(\alpha)$	$\cos(\alpha)$
R_1	rotacja	0,459700	0,888074
P_1	permutacja	-	-
R_2	rotacja	0,577349	0,816497
P_2	permutacja	-	-
R_3	rotacja	-0,888074	0,459700
R_4	rotacja	0,459700	0,888074
P_3	permutacja	-	-
R_5	rotacja	0,577349	0,816497
P_4	permutacja	-	-
R_6	rotacja	-0,888074	0,459700
R_7	rotacja	0,459700	0,888074
P_5	permutacja	-	-
R_8	rotacja	0,577349	0,816497
P_6	permutacja	-	-
R_9	rotacja	-0,888074	0,459700

W tabeli 1 przedstawione są odpowiednie wartości funkcji sinus i cosinus dla poszczególnych rotatorów. Cały system, zbudowany z rotatorów oraz bloków opóźniających, przedstawiony jest na rysunku 1. Wielkość bloków opóźniających D_i (rys. 1) zależy od rozmiaru przetwarzanego obrazu ($X \times Y \times Z$). Ponieważ rotatory mogą być realizowane za pomocą algorytmu CORDIC [14], cały system jest w pełni potokowy.

Prawidłowe działanie zaprojektowanego układu zweryfikowane zostało w kolejnym punkcie.

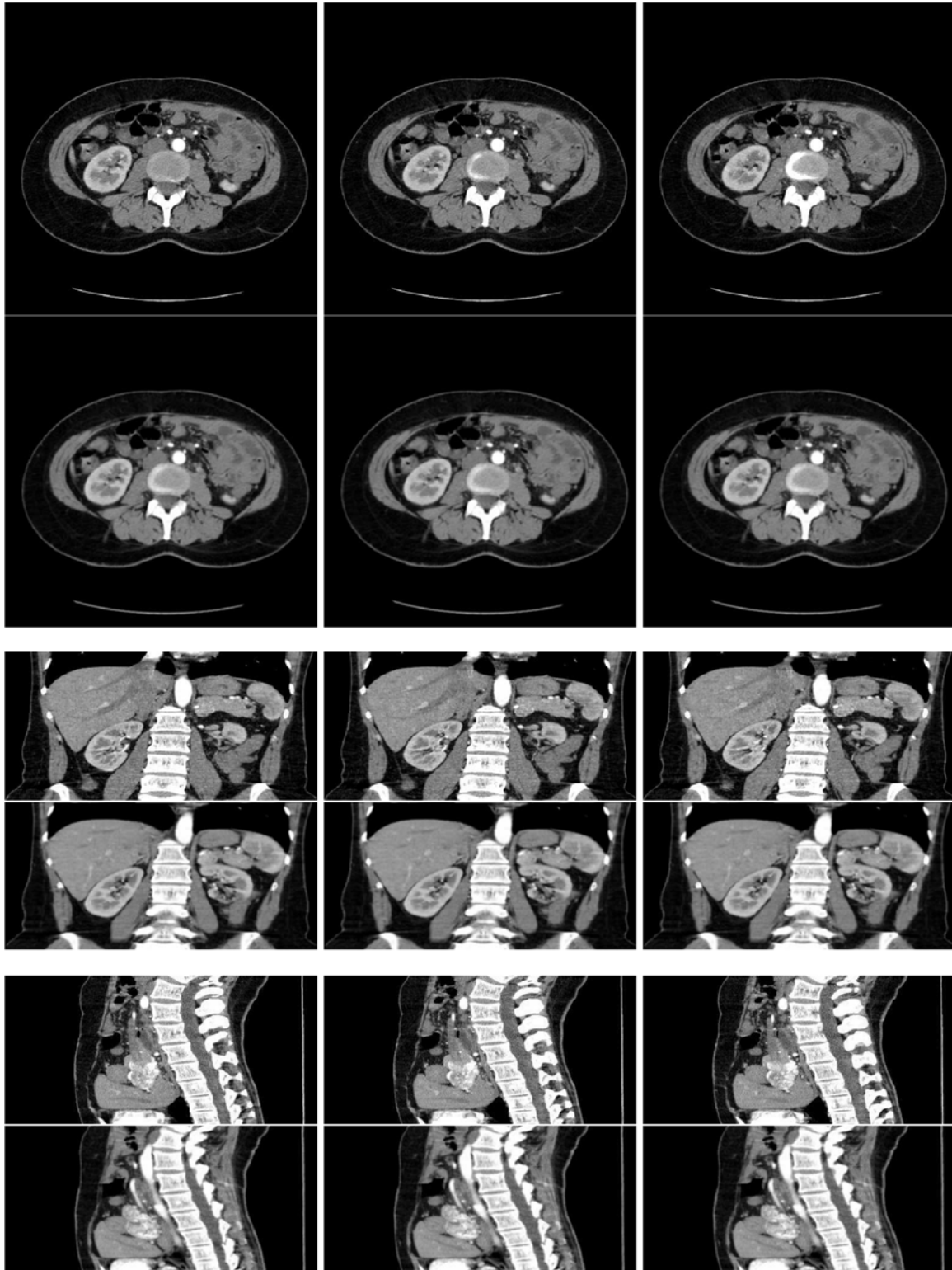


Rys. 1. Schemat blokowy filtra zrealizowanego za pomocą rotatorów i permutacji

Symulacja działania systemu oraz filtracja przykładowego obrazu

Symulacja systemu opartego na rotatorach i blokach opóźniających została wykonana również w środowisku Scilab. Dokonano pomiaru odpowiedzi impulsowej takiego układu na deltę Kroneckera, tj. macierz zerową o wymiarach $5 \times 5 \times 5$ z wartością 1 na pozycji początkowej ($h = 0, v = 0, d = 0$). Reakcją systemu na takie pobudzenie, są wartości pokrywające się ze współczynnikami filtru (3),

co potwierdza prawidłowe działanie filtru uśredniającego. W ramach testu przetworzony został przykładowy obraz medyczny 3D zapisany w formacie DICOM [15]. Obraz oryginalny oraz filtrowany potokową realizacją filtru uśredniającego 3D zaprezentowano na rysunku 2. Na przekrojach obrazu przetworzonego zauważyć można rozmycie (wygładzenie), które jest charakterystyczne dla dolnoprzepustowego filtru uśredniającego.



Rys. 2. Wybrane slajdy z obrazu oryginalnego (górną) i przetworzonego filtrem uśredniającym 3D (dół)

Podsumowanie

W artykule przedstawiona została potokowa realizacja filtra uśredniającego 3D w oparciu o rotatory i struktury opóźniające, z wykorzystaniem oryginalnych metod syntezy takiego systemu. Wyniki symulacji potwierdzają prawidłowe działanie układu, błąd średniokwadratowy odpowiedzi impulsowej wyniósł $5,140 \cdot 10^{-26}$ i wynika głównie z dokładności obliczeniowej systemu Scilab, która ma wpływ nie tylko na symulację, ale też na wszystkie etapy syntezy. Dzięki algorytmowi CORDIC nie ma konieczności wykorzystywania układów mnożących przy obliczaniu rotatorów. Ma to duży wpływ na implementację sprzętową, ponieważ pozwala na realizację takiego filtra w układach o mniejszej mocy obliczeniowej i bez stosowania specjalnych jednostek mnożących, które wykorzystywane są w metodach bezpośrednich. W efekcie daje to możliwość uzyskania w pełni potokowej struktury filtra, przy użyciu minimalnej ilości najprostszych elementów logicznych. Potokowość zapewni taką realizację przetwarzania, że kolejne próbki mogą być wprowadzane za każdym cyklem zegara, pomimo że struktura nie skończyła obliczeń próbek poprzednio wprowadzonych. Na wyjściu kolejne próbki, również pojawiają się za każdym cyklem zegara sterującego całym układem. Stanowi to główną zaletę względem układów tradycyjnych, gdzie wczytanie kolejnej próbki wejściowej możliwe jest dopiero po skończeniu obliczeń próbki poprzedniej. Zastosowanie rotatorów pozwoli jednocześnie na zmniejszenie wrażliwości układu na skończoną precyzję obliczeń. W dalszych etapach pracy, planowana jest implementacja sprzętowa z zastosowaniem struktur FPGA, co pozwoli wykorzystać wszystkie zalety systemu i dokładnie zbadać cały układ.

Autor: mgr inż. Paweł Poczekajło, Politechnika Koszalińska, Wydział Elektroniki i Informatyki, 75-453 Koszalin, ul. J.J. Śniadeckich 2, E-mail: pawel.poczekajlo@tu.koszalin.pl.

LITERATURA

- [1] Yoshida H., Toriwaki J., Fundamentals of Three-dimensional Digital Image Processing, Springer-Verlag, (2009)
- [2] Khali H., Riyadh M., Araar A., Real-time 3D image computation using LUT-based DSP systems, *The 2004 IEEE Asia-Pacific*

- Conference on Circuits and Systems*, (2004), vol. 1, 369-372
- [3] Kang Zhang, Jin U. Kang, Real-time 4D signal processing and visualization using graphics processing unit on a regular nonlinear-k Fourier-domain OCT system, *Optics Express*, Vol. 18, Issue 11 (2010), 11772-11784
- [4] Thomas C., DeVries P., Hardin J., White J., Four-Dimensional Imaging: Computer Visualization of 3D Movements in Living Specimens, *Science*, Vol. 273, No. 5275 (1996), 603-607
- [5] Dandekar O., Castro-Pareja C., Shekhar R., FPGA-based realtime 3D image preprocessing for image-guided medical interventions, *JRTIP*, 1 (2007), 285-301
- [6] Teßmann M., Mohr S., Gayetskyy S., Haßler U., Hanke R., Greiner G., Automatic Determination of Fiber-Length Distribution in Composite Material Using 3D CT Data, *EURASIP journal on advances in signal processing*, (2010), Art. 545030
- [7] Verdonck L., Vermeulen F., Docter R., Meyer C., Knies R., 2D and 3D ground-penetrating radar surveys with a modular system: data processing strategies and results from archaeological field tests, *Near Surface Geophysics*, Vol 11, No 2, April 2013, 239-252
- [8] Wirski R.T., Wawryn K., Stanowa synteza systemów bezstratnych o skończonej odpowiedzi impulsowej, *Przegląd Elektrotechniczny*, 86 (2010), nr.11a, 218-221
- [9] Wirski R.T., Wawryn K., Strzeszewski B., State-space approach to implementation of FIR systems using pipeline rotation structures, *ICSES*, Wrocław, (2012)
- [10] Poczekajło P., Wirski R.T., Synteza separowalnych trójwymiarowych filtrów ortogonalnych o strukturze potokowej, *Przegląd Elektrotechniczny*, 89 (2013), nr.10, 150-152
- [11] Wawryn K., Poczekajło P., Wirski R., FPGA implementation of 3-D separable Gauss filter using pipeline rotation structures, *MIXDES 2015, 22nd International Conference 25-27 June 2015*, 589-594
- [12] Ohser J., Schladitz K., 3D Images of Materials Structures: Processing and Analysis, Wiley-VCH, Weinheim, Germany (2009)
- [13] Home – Scilab [online], <http://www.scilab.org/>, data dostępu 15.01.2016r.
- [14] Volder J.E., The CORDIC trigonometric computing technique, *IRE Trans. Electron. Comput.*, vol. EC-8, no. 3, 330-334, Sep. 1959
- [15] DICOM files [online], <http://www.osirix-viewer.com/datasets/>, data dostępu 15.01.2016r.