

Agnieszka Siwocha*, Marcin Krupski*, Andrzej Cader*

Koncepcja fraktalnej kompresji obrazu

1. Wprowadzenie

Świat rzeczywisty wypełniają złożone obrazy o bardzo skomplikowanych kształtach i strukturze. Cechuje je jednak wysoki stopień redundancji, wynikający m.in. z własności samopodobieństwa [1]. Analizy statystyczne pokazują, że większość obrazów naturalnych daje się rozdzielić na części, które są fraktalami. Dają się więc zapisać w postaci systemu funkcji iterowanych IFS (*Iterated Function System*) [2], co w znakomity sposób zmniejsza ilość niezbędnej do przechowania informacji potrzebnej do odtworzenia obrazu. Metoda kompresji obrazu opierająca się na IFS została opracowana przez Barnsleya i współpracowników [3–7] oraz rozwinięta w wielu kierunkach [8–14]. Bardziej zaawansowane metody traktują naturalne obrazy jak obiekty multifraktalne [15, 16]. IFS jest zbiorem n afinicznych przekształceń zwięzających przekształcających obraz A

$$W(A) = w_1(A) \cup w_2(A) \cup \dots \cup w_n(A) \quad (1)$$

Ten zbiór przekształceń nosi też nazwę operatora Hutchinsona. Udowadnia się, że ciąg kolejnych obrazów uzyskanych zależnością rekurencyjną $A_{i+1} = W(A_i)$ jest zbieżny (niezależnie od wyboru A_0) do granicy A_∞ będącej atraktorem – punktem stałym danego IFS: $A = W(A)$. Ponieważ za A_0 można wybrać dowolny obraz, obraz kodowany daje się jednoznacznie zapisać za pomocą współczynników przekształceń afinicznych w ilości $6n$. Teoretycznie atraktor uzyskuje się po nieskończonej ilości iteracji, jednak w praktyce wystarczy już kilkanaście iteracji. Podstawowym problemem w metodzie IFS stanowi określenie operatora Hutchinsona dla danego obrazu. Podstawowa metoda polega na wykorzystaniu faktu, że samopodobieństwo może występować pomiędzy fragmentami obrazu. Obraz jest wtedy dzielony na rozłączne części R_i – domeny docelowe $R_i \cap R_j = \emptyset$ dla $i \neq j$ pokrywające cały obraz $\cup R_i = A$. Dla każdego podzbioru R_i znajdująca jest dziedzina D_i – domena źródłowa ($D_i \supset R_i$), taka że istnieje odwzorowanie zwięzające, dla którego $w_i(D_i) = R_i$. Zbiór odwzorowań tworzy operator, którego punktem stałym (atraktorem) jest oryginalny obraz. Zwykle

* Instytut Kształcenia na Odległość, Wydział Informatyki, Zarządzania i Transportu, Wyższa Szkoła Humanistyczno-Ekonomiczna w Łodzi

odwzorowania odtwarzają obraz jedynie z pewnym przybliżeniem. Należy zauważyć, że każde z powyższych odwzorowań działa jedynie na pewien fragment obrazu, a nie na cały obraz.

Podstawową grupą metod znajdowania systemu odwzorowań kodujących obraz są metody polegające na dzieleniu oryginalnego obrazu na części (R_i) według różnych reguł: równomierne obszary, rozbitcie poczwórne, rozbitcie HV, rozbitcie trójkątne itp. Następnie wybiera się zbiór dziedzin przekształceń (D_i) jako zbiór wszystkich możliwych obszarów o zadanych wymiarach i metodą najmniejszych kwadratów, w której minimalizuje się odległość w metryce średniokwadratowej pomiędzy obrazem uzyskanym za pomocą sugerowanego operatora a obrazem oryginalnym, wyznacza się zbiór współczynników poszczególnych przekształceń. Metoda ta jest niezmiernie czasochłonna i w dużej mierze zależy od wyboru *a priori* R_i oraz D_i . W niniejszej pracy zaproponowano odmienną koncepcję kompresji fraktalnej obrazów, w założeniach różniącą się zasadniczo od metody IFS. Proponuje się wykorzystanie nowej metody interpolacji fraktalnej przedstawionej w pracach [17] i [21]. Metoda ta opiera się na koncepcji fraktalnych splajnów bazowych (*FB-splines*) i jest modyfikacją klasycznej metody splajnów bazowych. W niniejszej pracy metoda FBS została zaproponowana jako tymczasowe narzędzie, w pracach [17] i [21] nie przedstawiono takiego zastosowania metody. Należy podkreślić, że wszystkie metody kompresji oparte na fraktalach cechuje wyjątkowo wysoki, nie dający się osiągnąć innymi metodami współczynnik kompresji. Wynika to z samej natury fraktali.

2. Podstawy teoretyczne metody

W proponowanej metodzie przyjmuje się, że wartości jasności kolejnych pikseli w linii obrazu tworzą szereg będący wynikiem próbkowania złożonej struktury o charakterze multifraktala, będącej odzwierciedleniem rzeczywistego obrazu. Można więc do takiego szeregu zastosować znane metody analizy oparte na badaniu widma multifraktalnego [18, 19]. Taki szereg jest zbiorem numerycznych wartości, z których każdą można traktować jak wynik realizacji pewnego procesu stochastycznego $P(x)$. Wartości w wyrażonym szeregu (T_1, T_2, \dots, T_N) określamy dla jednakowych przedziałów Δx , gdzie

$$T_i = P(x = i\Delta x) \quad (2)$$

Dla ułatwienia, szereg jest znormalizowany

$$\frac{1}{N} \sum_{k=1}^N T_k = 1 \quad (3)$$

Definiuje się funkcję ziarnistości

$$T(x) = \sum_{k=1}^N T_k I_{[(k-1)\Delta x, k\Delta x]}(x) \quad (4)$$

gdzie $I_A(x)$ jest charakterystyczną funkcją zbioru A postaci:

$$I_A(x) = \begin{cases} 1, & \text{dla } x \in A \\ 0, & \text{dla } x \notin A \end{cases} \quad (5)$$

Analiza multifraktalna polega na znalezieniu nowego szeregu $S(x)$ złożonego z liczb całkowitych σ_i wybranych z przedziału $1, 2, \dots, Q \ll N$, gdzie każda z tych liczb odpowiada wartości ziarnistego wykładnika Holdera (*coarse Holder exponent*) z przedziału $[(r-1)\Delta x, r\Delta x]$ – forma dyskretyzacji wykładnika Holdera α

$$S(x) = \sum_{k=1}^N \sigma_k I_{[(k-1)\Delta x, k\Delta x]}(x) \quad (6)$$

Taki sposób postępowania ma na celu wyodrębnienie podzbiorów składających się z przedziałów o takiej samej wartości σ_i (dyskretnej wartości wykładnika Holdera). Podzbiory te tworzą składniki tzw. optycznego widma multifraktalnego. Często wyraża się je graficznie, przypisując jeden kolor dla przedziałów o tej samej wartości σ_i [20]. Dla poszczególnych podzbiorów wylicza się wymiar fraktalny D_i . Widmo multifraktalne tworzy zależność $D(\alpha)$.

Z powyższego widać, że w analizie multifraktalnej podstawowe znaczenie ma pojęcie wykładnika Holdera. Niech będzie dana zmienna losowa ζ . Prawdopodobieństwo P , że zmienna ta znajduje się w przedziale δx wokół punktu x , określa miarę

$$\mu(B_{\delta x}(x)) = P \left\{ x - \frac{\delta}{2} < \zeta < x + \frac{\delta}{2} \right\} \quad (7)$$

Może to być np. prawdopodobieństwo, że przypadkowo wybrany punkt krzywej znajduje się w obszarze $B_{\delta x}(x)$. Odpowiednią miarę, tak zwaną miarę Billingsleya [25] dla każdego punktu x definiuje wyrażenie

$$G_d(x) = \lim_{\delta x \rightarrow 0} \mu(B_{\delta x}(x))(\delta x)^{-d} \quad (8)$$

W ogólności $G_d(x)$ dla dowolnej wartości d przyjmuje wartość zero lub nieskończoność, istnieje tylko jedna wartość $d = \alpha(x)$, dla której miara Billingsleya jest skończona. Wartość ta określa wykładnik Holdera w punkcie x :

$$\begin{aligned} G_d(x) &= 0, \text{ gdy } d < \alpha(x) \\ G_d(x) &= \text{jest skończone, jeżeli } d = \alpha(x) \\ G_d(x) &= \infty, \text{ gdy } d > \alpha(x) \end{aligned} \quad (9)$$

W przypadku gdy rozkład punktów jest singularny, a ma to miejsce w przypadku zbiorów fraktalnych, wykładnik Holdera $\alpha(x)$ określa wielkość tej singularności. Wybierając zbiór punktów krzywej, dla których wykładnik Holdera ma taką samą wartość $H_{\alpha_i} = \{x: \alpha(x) = \alpha_i\}$, można wyznaczyć wymiar fraktalny (wymiar Hausdorffa) tego zbioru $D(\alpha) = Dim(H_\alpha)$. Zależność ta nazywa się widmem multifraktalnym.

W praktyce mamy do czynienia ze strukturami dyskretnymi przedstawianymi z określoną rozdzielczością, np. w algorytmach komputerowych uwarunkowanymi wielkością piksela. W takim przypadku stosuje się ziarnisty wykładnik Holdera. Dla przedziału δx wokół punktu x definiuje się $\alpha(x, \delta x)$ za pomocą zależności

$$G_{\alpha(x, \delta x)}(x) = \mu(B_{\delta x}(x))(\delta x)^{-\alpha(x, \delta x)} \quad (10)$$

$\alpha(x, \delta x)$ można wyznaczyć, logarytmując powyższą zależność

$$\ln \mu(B_{\delta x}(x)) = \alpha(x, \delta x) \ln \delta x + \ln G_{\alpha(x, \delta x)}(x) \quad (11)$$

i stosując znaną procedurę polegającą na wyznaczaniu punktów we współrzędnych $(\ln \mu(B_{\delta x}(x)), \ln(\delta x))$ dla przedziałów o różnej wielkości. Ziarnisty wykładnik Holdera określa nachylenie linii prostej dopasowanej metodą najmniejszych kwadratów. Procedurę należy zastosować do każdego punktu (piksela).

3. Zasady proponowanej metody kompresji obrazów

W interpolacyjnych metodach fraktalnych znaczącą rolę odgrywa pojęcie węzła interpolacyjnego. Jednak w przypadku odtwarzania złożonych struktur fraktalnych klasyczne pojęcie węzła interpolacyjnego przestaje wystarczać, gdyż nie można mu przypisać jednej, jednoznacznej wartości. Pojawia się konieczność wprowadzenia nowej wielkości określającej miejsce pobierania informacji, za pomocą której w przyszłości struktura będzie odtwarzana. Takim nowym pojęciem jest lokalne okno interpolacyjne LIW (*Local Interpolation Window*) [17, 21]. LIW odgrywa zasadnicze znaczenie dla pozyskania informacji potrzebnej do lokalnego odtworzenia interpolowanego kształtu. Liczbowe dane z zaznaczonego obszaru LIW, takie jak lokalny wymiar fraktalny, tworzą zbiór danych, na podstawie których rekonstruowana jest krzywa reprezentująca zależność jasności od położenia piksela. Dobór powyższych parametrów, oprócz określenia jakości interpolacji, stanowi również o stopniu kompresji. Takie podejście do interpolacji krzywych fraktalnych wymaga badania wymiaru fraktalnego krzywych, a w przypadku fraktali stochastycznych – widma multifraktalnego.

Stosując podejście przybliżone, można ziarnisty wykładnik Holdera $\alpha(x, \delta x)$ uśredniać przedziałami, reprezentującymi szerokość LIW

$$\alpha(x) = \sum_{i=1}^N \langle \alpha(x, \delta x) \rangle_{\Delta x_i} I_{\Delta x_i}(x) \quad (12)$$

gdzie:

$I_{\Delta x_i}(x)$ – funkcja charakterystyczna (5),

$\langle \alpha(x, \delta x) \rangle_{\Delta x_i}$ – wartość średnia $\alpha(x, \delta x)$ w przedziale Δx_i .

W wyniku tego krzywa multifrakalna rozpada się, z pewnym przybliżeniem zależnym od szerokości LIW, na lokalne (w przedziałach Δx_i) zbiory monofrakalne o jednej unikalnej wartości wymiaru fraktalnego $D(\Delta x_i) = D_i$ w każdym przedziale. Pierwotna krzywa może być wtedy odtwarzana lokalnie za pomocą zadanych krzywych fraktalnych o odpowiednim wymiarze. W praktyce wymiar w tym przypadku może być obliczany dla każdego okna jak wymiar pudełkowy, co znacznie ułatwia obliczenia.

W celu zwiększenia dokładności interpolacji można przyjąć większą liczbę dyskretnych wartości wykładnika Holdera w oknie interpolacyjnym. Niech

$$\Delta \alpha_i = \frac{(\alpha_i(x)_{\max} - \alpha_i(x)_{\min})}{L-1} \quad (i - \text{numer okna, } j = 1, \dots, L).$$

Wtedy fragment krzywej należącej do danego okna rozpada się na $L - 1$ podzbiorów S_{ij} takich, że

$$S_{ij} = \{x : \alpha(x) \in [\alpha_i(x)_{\min}, \alpha_i(x)_{\min} + j\Delta \alpha_i]\} \quad \text{dla } j = 1, \dots, L-1.$$

Każdy z tak określonych podzbiorów ma swój unikalny wymiar fraktalny $D_{ij} = \text{Dim}(S_{ij})$. W ten sposób fragment krzywej z danego LIW został potraktowany jak dyskretny multifraktal, który może być odtwarzany za pomocą $L - 1$ krzywych o odpowiednich wymiarach.

Do odtworzenia interpolowanej struktury niezbędny jest zbiór krzywych bazowych \mathbf{K} będących fraktalami określonymi na standardowym przedziale Δx , np. $\Delta x = [0, 1]$. Ze względu na konieczność dysponowania fraktalami bazowymi o dowolnym wymiarze fraktalnym z przedziału $D \in [1, 2)$, zbiór \mathbf{K} musi być zupełny. Wymagamy ponadto by operacja skalowania nie zmieniała wymiaru fraktalnego krzywych bazowych.

Spośród wielu możliwych do zdefiniowania zbiorów krzywych fraktalnych spełniających powyższe warunki, do praktycznego wykorzystania wybrano rodzinę krzywych opartych na dobrze znanej krzywej Kocha, krzywe te będziemy nazywać uogólnionymi krzywymi Kocha GKC (*Generalized von Koch Curve*) [17]. Zaletą takiego wyboru jest prosta zależność określająca wymiar fraktalny GKC

$$D = \frac{\ln 4}{\ln [2(1 + \cos \varphi)]} \quad (13)$$

w powyższym parametrze $\varphi \in [0, \frac{\pi}{2})$.

Zbiór \mathbf{K} jest wykorzystany w algorytmie interpolacyjnym. Załóżmy, że dane są wartości krzywej y_1, y_2, \dots, y_n w zadanych węzłach interpolacyjnych x_1, x_2, \dots, x_n (reprezentują one jasności wyróżnionych pikseli) oraz wymiary fraktalne D_1, D_2, \dots, D_n interpolowanej krzywej L wyznaczone w poszczególnych oknach interpolacyjnych LIW odpowiadających danym węzłom (krok interpolacyjny jest stały i równy Δx). Dla uproszczenia, w rzeczywistych obliczeniach stosuje się wymiar pudełkowy (pojemnościowy)

$$D = \lim_{\epsilon \rightarrow 0} \frac{\log N(\epsilon)}{\log(1/\epsilon)} \quad (14)$$

gdzie N jest minimalną liczbą kostek o boku ϵ potrzebnych do pokrycia badanego zbioru fraktalnego.

Interpolowaną krzywą może wyznaczyć za pomocą zależności

$$L \approx \sum_{i=0}^N s_i K_i(x - i\Delta x, y, D_i) \quad (15)$$

gdzie s_i współczynniki interpolacji, a krzywe fraktalne $K_i(x, y)$ umieszczone są centralnie (z wartością maksymalną) w punktach interpolacji.

W powyższym wyrażeniu należy zwrócić uwagę na to, że w odróżnieniu od klasycznej metody *B-splines*, w której skalowanie następuje tylko względem osi y , tutaj krzywe skalowane są we wszystkich kierunkach – konieczność zachowania wymiaru fraktalnego. Ponadto sumowanie określone zostało symbolicznie jako sumowanie krzywych fraktalnych, natomiast sposób tego sumowania dokładnie został zdefiniowany w [17, 21] i przeprowadzany jest za pomocą specjalnego algorytmu. Współczynniki s_i wylicza się iteracyjnie na podstawie wyrażenia

$$y_i = \sum_{j=0}^N s_j^{n+1} y_j \left(\frac{(i-j)\Delta x}{s_j^n}, D_j \right)^* \quad (16)$$

gdzie:

$s_j^0 = 1, y_i$ – zadane wartości węzłów interpolacyjnych,

$y_j(x, D_j)$ – zbiór wartości y krzywej $K_j(x, y, D_j)$ o numerze j dla zadanej wartości x , przy czym $K_j(x, y, D_j)$ to startowe FB-splajny ułożone są centralnie w każdym punkcie interpolacyjnym.

Do obliczeń przyjmowano rozpiętość krzywych bazowych równą $6\Delta x$. W wyrażeniu (16) uwzględniono możliwość wystąpienia wielu wartości y FB-splajnow przy ustalonym x . Stąd

$$y_j^n(x, D_j)^* = \frac{1}{2} \left[\sup y_j^n(x, D_j) + \inf y_j^n(x, D_j) \right] \quad (17)$$

Miarą dokładności obliczeń jest średni błąd kwadratowy iteracji

$$\delta^n = \sum_{i=0}^N \left(y_i - \sum_{j=0}^N y_j \left((i-j)\Delta x, D_j \right)^* \right)^2 \quad (18)$$

Przy interpolacji należy uwzględnić dobrze znany problem braku danych na końcach interpolowanej krzywej. Wartości te należy zdefiniować dodatkowo.

4. Algorytm metody

Poniżej przedstawiono algorytm opisanej metody kompresji. Aplikacja buduje, na podstawie danych pobranych z obrazu przeznaczonego do kompresji, wykres jasności kolejnych linii obrazu. W pierwszym etapie użytkownik dokonuje wyboru liczby węzłów interpolacyjnych. Program będący implementacją opisywanego algorytmu odczytuje liczbę węzłów z interfejsu użytkownika i buduje tablice wartości rzędnych $x[w]$ i odciętych $y[w]$ dla poszczególnych węzłów. Kolejnym etapem jest określenie rozmiarów lokalnych okien interpolacyjnych dla poszczególnych węzłów. Z LIW pobierane będą charakterystyki nielokalne. Przyjmowana jest domyślna wartość szerokości okna równa krokowi interpolacyjnemu. Dzięki temu okna nie zachodzą na siebie oraz nie występuje strefa martwa pomiędzy oknami. Obliczanie wysokości (minimalnej i maksymalnej wartości współrzędnej y) okna odbywa się w pętli po szerokości okna (po współrzędnej x), w tym miejscu posłużono się standardowym algorytmem wyszukiwania maksimum i minimum (rys. 1).

Po wyznaczeniu LIW, wykonywana jest część programu odpowiedzialna za obliczenie charakterystyk dla okien w poszczególnych węzłach – wyznaczany jest wymiar pojemnościowy. Graficzna zawartość okna interpolacyjnego kopiowana jest do binarnej tablicy o analogicznych rozmiarach. W pętli obliczającej wartość wymiaru pudełkowego jest on obliczany dla trzech wartości wymiaru kostki ϵ . Licznikiem pętli jest ϵ przyjmująca kolejno wartości 4, 2 i 1.

Zmienna $N(\epsilon)$ przyjmuje wartość równą liczbie bloków o boku ϵ , w których obszarze znalazł się co najmniej jeden piksel zaczerniony wykresem jasności (rys. 2). Trzy odczyty wartości ϵ i $N(\epsilon)$ interpretowane są za pomocą metody najmniejszych kwadratów, a otrzymane nachylenie prostej określa wymiar fraktalny D w rozpatrywanym oknie (rys. 3). Tak obliczane wymiary fraktalne ze wszystkich okien dla zadanych węzłów umieszczone są w jednowymiarowej tablicy $D[w]$.

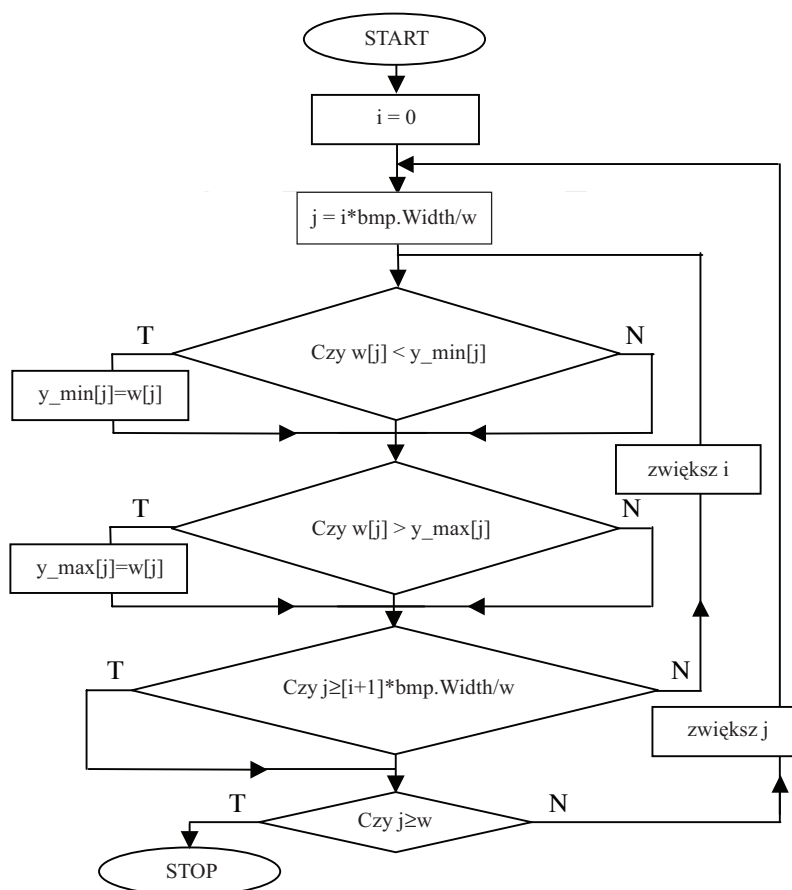
Po zgromadzeniu niezbędnych danych (tablice $x[]$, $y[]$, $D[]$) tworzony jest obiekt usługowej metody FBS. Dla obiektu FBS należy zainicjować pola reprezentujące współrzędne i wymiary fraktalne dla zadanych węzłów interpolacyjnych. W tym miejscu aplikacja podaje tablice $x[]$, $y[]$ i $D[]$.

Zdarzenia wywołane przez odpowiednie elementy interfejsu powodują kolejno:

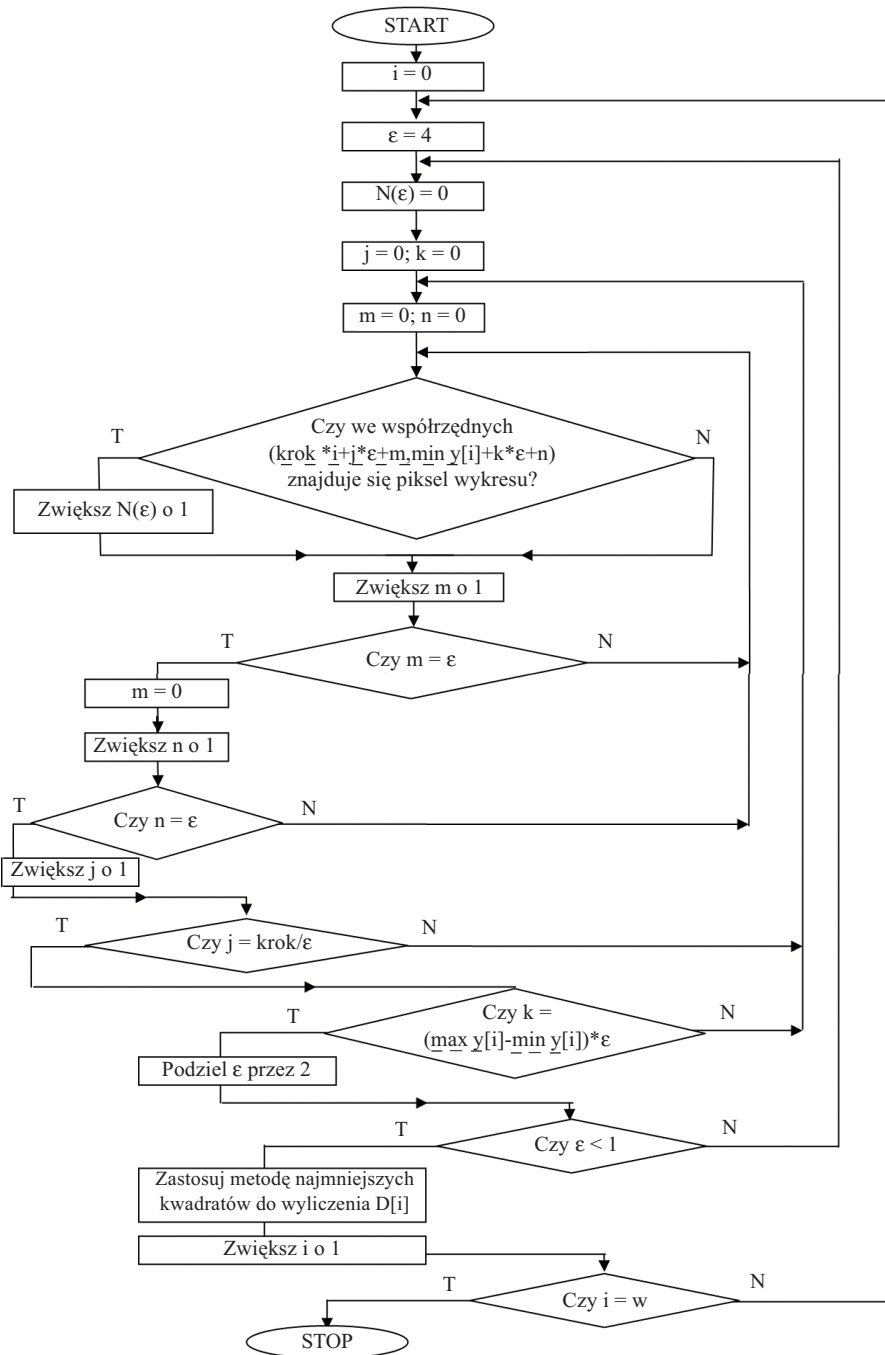
1. Wywołanie metody *FBS.TworzKK* – odpowiada za zbudowanie fraktalnych splajnów bazowych o domyślnej długości, rozpiętych symetrycznie względem zadanych wartości x i posiadających odpowiedni wymiar fraktalny D .

2. Wywołanie najważniejszego modułu stanowiącego istotę algorytmu FBS: metody *FBS.Interpoluj* obliczającej w sposób iteracyjny współczynniki interpolacji (tablica współczynników $s[n]$) dla przeskalowania utworzonych uprzednio fraktalnych splajnow bazowych.
3. Uruchomienie metody *FBS.Sumuj* – metoda sumuje przeskalowane uogólnione krzywe Kocha tworzące krzywą wynikową.

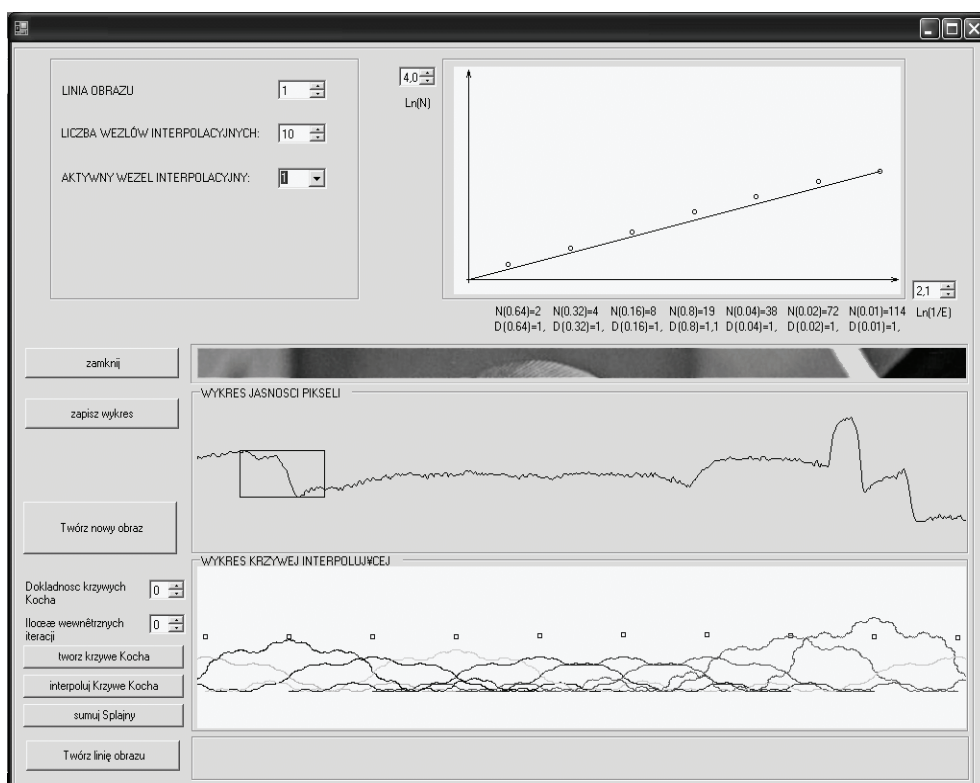
Powyższe metody szczegółowo opisane zostały w [21]. Do odtworzenia obrazu przechowywane są informacje o wartości kroku interpolacyjnego oraz wymiary fraktalne uogólnionych krzywych Kocha (tablica $D[w]$) i współczynniki skalowania poszczególnych splajnow (tablica $s[w]$). Na podstawie tych danych można bez trudu odtworzyć obraz. Przy odtwarzaniu za pomocą metody *FBS.TworzKK* tworzone są ponownie FB-splajny w punktach interpolacji, następnie są one skalowane w oparciu o znane wartości $s[w]$ i sumowane do krzywej wynikowej metodą *FBS.Sumuj*. Na podstawie odtworzonej krzywej, czyli zrekonstruowanego wykresu jasności, linia po linii budowany jest obraz.



Rys. 1. Schemat blokowy algorytmu służącego do wyznaczenia położenia i rozmiarów okna LIW



Rys. 2. Schemat blokowy algorytmu wyznaczającego wymiary fraktalne $D[i]$ w oknie LIW



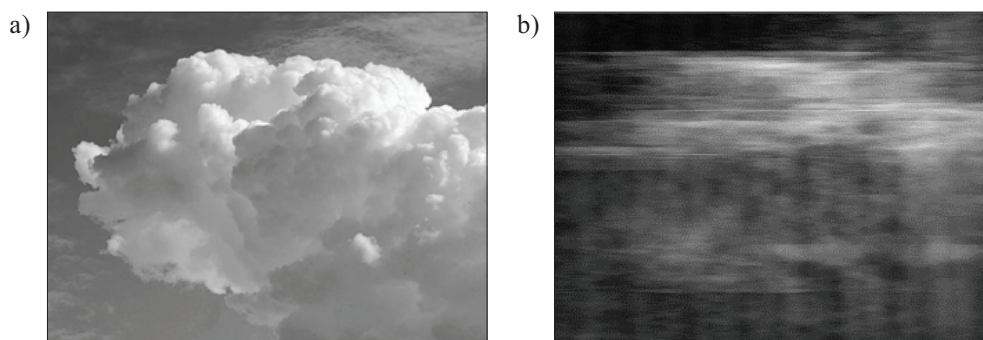
Rys. 3. Ilustracja działania algorytmu znajdującego LIW (przy zadanych 10 węzłach interpolacyjnych) i wyznaczającego wymiar fraktalny D dla LIW w drugim węzle interpolacyjnym dla jednej linii obrazu

5. Podsumowanie

Pierwsze przeprowadzone badania wskazują na duże możliwości zaproponowanej metody, zwłaszcza w zakresie możliwości uzyskiwania bardzo wysokich stopni kompresji. Już przy dziesięciu, dwudziestu węzłach interpolacyjnych na linię (800 pikseli) najprostsza wersja metody (opisana powyżej) z jednym wymiarem fraktalnym dla okna interpolacyjnego daje zauważalne rezultaty. Tak wysoki stopień kompresji jest możliwy dzięki fraktalnym właściwościom splajnów odtwarzających (rys. 4).

Należy zwrócić uwagę na fakt, że metoda jest metodą rozwojową. Trwają prace nad bardziej złożonym modelem charakterystyk opisujących strukturę krzywych w oknie interpolacyjnych. Niektóre kierunki badań zostały wskazane w pracy [17]. Ponadto w opisanym algorytmie wprowadzić należy pełniejszy wachlarz wymiarów fraktalnych dla LIW. W związku z tym można oczekiwać znacznego polepszenia jakości kompresji – mniejszej stratności. Sprawą otwartą jest dalszy rozwój metody FBS, szczególnie, jeśli chodzi o wy-

bór charakterystyk nielokalnych z otoczenia węzła interpolacyjnego. Odpowiednie dobranie takich charakterystyk i ich celne wykorzystanie może mieć znaczący wpływ na poprawę jakości odtwarzanego obrazu.



Rys. 4. Efekt działania algorytmu; obrazy: a) przed kompresją; b) odtworzony po kompresji

Literatura

- [1] Turiel A., Parga N.: *The multi-fractal structure of contrast changes in natural images: from sharp edges to textures*. Neural Computation, vol. 12, 763–793
- [2] Hutchinson J.E.: *Fractals and Self-Similarity*. Ind. Univ. J. Math., 30, 1981, 713–747
- [3] Barnsley M.F.: *Fractals Everywhere*. 2nd ed., Academic Press, Inc., Boston, 1993
- [4] Barnsley M.F., Demko S.: *Iterated function systems and the global construction of fractals*. Proc. R. Soc. London, A 399, 1985, 243–275
- [5] Barnsley M.F.: *Fractal image compression*. Notices AMS, 43, 1996, 657–662
- [6] Barnsley M.F., Hurd L.P.: *Fractal Image Compression*. A.K. Peters, Boston, MA, 1992
- [7] Jacquin A.: *Image coding based on a fractal theory of iterated contractive image transformations*. IEEE Trans. Image Processing, 1, 1992, 18–30
- [8] Brammer R.F.: *Unified image computing based on fractals and chaos model techniques*. Optical Eng., 28, 1989, 726–734
- [9] Dubuc S., Elqortobi A.: *Approximations of fractal sets*. J. Comput. Appl. Math., 29, 1990, 79–89
- [10] Fisher Y.: *Fractal Image Compression*. Springer, Berlin, 1995
- [11] Goodman G.S.: *A probabilist looks at the chaos game*. In Peitgen H.O. et al. (eds), Fractals in the Fundamental and Applied Sciences, Proc. 1st IFIP Conf. on Fractals, Lisbon, 1990, 1991, 159–168. Elsevier
- [12] Hepting D., Prusinkiewicz P.: *Rendering methods for iterated function systems*. In Peitgen, H. O. et al. (eds) Fractals in the Fundamental and Applied Sciences, Proc. 1st IFIP Conf. on Fractals, Lisbon, 1990, 1991, 183–224. Elsevier
- [13] Hoggar S.G., McFarlane I.: *Optimal sequences for non-uniform iterated function systems*. Bulletin Inst. Combin. Applic., 12, 1994, 65–92
- [14] Monro D.M., Wilson D.L., Nicholls J.A.: *High speed image encoding with the bath fractal transform*. In Proc. IEEE Symp. Multimedia Technologies and Future Applications, Signal Processing Chapter, Southampton, April 1993
- [15] Turiel A., Mato G., Parga N., Nadal J.P.: *The self-similarity properties of natural images resemble those of turbulent flows*. Physical Review Letters, vol. 80, 1998, 1098–1101

-
- [16] Turiel A, Parga N., Ruderman D., Cronin T.: *Multiscaling and information content of natural color images*. Physical Review E, vol. 62, 2000, 1138–1148
 - [17] Cader A., Krupski M.: *New interpolation method with fractal curves*. In: Artificial Intelligence and Soft Computing. Lecture Notes in Artificial Intelligence. Springer, 2006
 - [18] Halsey T.C., Jensen M.H., Kadanoff L.P., Procaccia I., Shraiman B.I.: *Fractal measures and their singularities: The characterization of strange sets*. Phys. Rev. A **33**, 1986, 1141
 - [19] Voss R.: *Random fractals: characterisation and measurement*. In Scaling Phenomena in Disordered Systems, edited by R. Pynn and A. Skjeltorp, New York: Plenum Press, 1986, 1–11
 - [20] del Rio Correa J.L., Muñoz-Diosdado A.: *Multifractality in Physiological Times Series*. Medical Physics, Sixth Mexican Symposium, 2002
 - [21] Krupski M., Siwocha A., Cader A.: *Interpolacja fraktalna w grafice komputerowej - wykorzystanie nowej metody*. Przetwarzanie i analiza sygnałów w systemach wizji i sterowania, Słok k/Belchatowa, 2006