

Adam BOROWICZ

Politechnika Białostocka, Wydział Informatyki,
ul. Wiejska 45A, 15-351 Białystok
E-mail: a.borowicz@pb.edu.pl

Zastosowanie spaczonych zbiorów Julii w symulacji tekstur nieba

1 Wstęp

Systemy rzeczywistości wirtualnej (ang. *virtual reality - VR*) zyskują w ostatnich latach coraz większą popularność. Ich kluczowym elementem jest wirtualne środowisko, stworzone dzięki komputerowej symulacji zjawisk zachodzących w rzeczywistym świecie. Środowisko to generowane jest najczęściej z wykorzystaniem technik grafiki 3D. Zwykle odwzorowuje ono pewien obszar otwarty, którego głównymi składnikami są: teren, roślinność, zbiorniki wodne i oczywiście niebo. W niektórych systemach VR obrazowanie nieba jest szczególnie ważne. Najlepszym tego przykładem są symulatory lotu [1]. Techniki te znajdują też szerokie zastosowanie w zwykłych grach komputerowych, szczególnie w zyskujących coraz większą popularność futurystycznych symulatorach kosmicznych. W tym ostatnim przypadku problem jest jednak bardziej złożony, gdyż w istocie dotyczy obrazowania nieba nocnego, którego głównymi elementami są ciała niebieskie, gwiazdozbiory, czy też całe galaktyki.

Najprostszą techniką graficznej symulacji nieba jest odwzorowanie, uprzednio przygotowanych, tekstur na siatce wielokątów reprezentujących półsferę (ang. *skydome*) [2]. Jest to podejście z jednej strony czasochłonne, gdyż wymaga udziału artystów grafików w przygotowaniu odpowiednich tekstur, a z drugiej strony pochłaniające znaczne zasoby pamięciowe kart graficznych. W celu osiągnięcia dużego stopnia realizmu wymagane są bowiem obrazy nieba o dużej rozdzielczości.

W przypadku metod proceduralnych [3] wykorzystuje się najczęściej generatory liczb pseudolosowych. Tekstury chmur generowane są automatycznie poprzez odpowiednią kompozycję obrazów szumu (oktaw) w różnej skali. Najczęściej odbywa się to w trakcie renderingu, niezależnie dla każdego piksela, co pozwala na zrównoleglenie obliczeń i obniżenie złożoności pamięciowej. Wygenerowane w ten sposób struktury są dosyć realistyczne, posiadają też pewne cechy zbiorów fraktalnych, w szczególności są one samopodobne w sensie stochastycznym. Niestety, techniki opierające się wyłącznie na generatorach liczb pseudolosowych nie pozwalają na uzyskanie bardziej wyrafinowanych kształtów, takich jak obrazy gwiazdozbiorów czy galaktyk.

W niniejszej pracy proponujemy wykorzystanie zbiorów Julii [4-5] do symulacji obrazów wirtualnych galaktyk. Reprezentacje graficzne tych zbiorów zawierają między innymi nietrywialnie struktury o kształtach spiralnych. W oryginalnej postaci wykazują one pewne symetrie. Jednakże z uwagi na fakt, że są określone w przestrzeni zespolonej, możliwa jest ich prosta deformacja (spaczenie) przy użyciu transmitancji filtru wszechprzepustowego. Po dodaniu do tak powstałych struktur turbulencji

o niewielkiej amplitudzie uzyskujemy realistycznie wyglądające obrazy skupisk gwiazd.

2 Fraktale i definicja zbioru Julii

Fraktal jest pojęciem stosunkowo nowym, które nie posiada uznanej jednoznacznie formalnej definicji. Najczęściej tym mianem określa się zbiór, posiadający szereg cech charakterystycznych, z których za najważniejsze uważa się samopodobieństwo oraz ułamkową wymiarowość. O ile samopodobieństwo jest cechą intuicyjnie zrozumiałą, o tyle pojęcie wymiarowości wymaga uściślenia. W ogólnym przypadku wymiarowość jest miarą samopodobieństwa i jednocześnie skomplikowania danego obiektu. Niestety, jest ona różnie definiowana. Istnieje zatem wymiar samopodobieństwa, pudełkowy, cyrkłowy, topologiczny [4] (s. 53). Jednak w przypadku większości obiektów fraktalnych miary te przyjmują wartości niecałkowite.

Jednym z bardziej popularnych obiektów fraktalnych jest zbiór Julii. Zakładając, że $f(z)$ to dowolna funkcja wymierna, wypełniony (ang. *filled-in*) zbiór Julii jest definiowany następująco [5] (s. 215):

$$K(f) = \{z \in \mathbb{C} : f^{(k)}(z) \not\rightarrow \infty, \text{ gdy } k \rightarrow \infty\}, \quad (1)$$

gdzie $f^{(k)}(z)$ oznacza k -krotne złożenie funkcji $f(z)$, tj.:

$$f^{(k)}(z) = f(f(f(\dots))) = \underbrace{f \circ f \circ \dots \circ f}_k. \quad (2)$$

Innymi słowy, zbiór ten tworzą punkty z , dla których ciąg iteracji wielomianu $f(z)$ ma ograniczoną orbitę (kolejne wartości $f^{(k)}(z)$ dla $k \rightarrow \infty$ nie „uciekają” do nieskończoności). Punkty te nazywane są też więźniami, a pozostałe uciekinierami. Natomiast „właściwy” zbiór Julii jest formalnie definiowany jako punkty brzegowe zbioru $K(f)$:

$$J(f) = \partial K(f). \quad (3)$$

Najdokładniej zbadane zbiory Julii opierają się na wielomianie drugiego stopnia:

$$f_c(z) = z^2 + c, \quad (4)$$

gdzie c - dowolny parametr zespolony. Dla różnych c otrzymuje się różne zbiory, dlatego w istocie mamy do czynienia z całą rodziną zbiorów Julii. Warto też nadmienić, że w zbiorze parametrów c definiowany jest popularny zbiór Mandelbrota:

$$M = \{c \in \mathbb{C} : f_c^{(k)}(0) \not\rightarrow \infty\}. \quad (5)$$

Zwykle zbiory Julii obrazowane są na płaszczyźnie zespolonej w wyniku przekształceń iteracyjnych wielomianu stopnia drugiego:

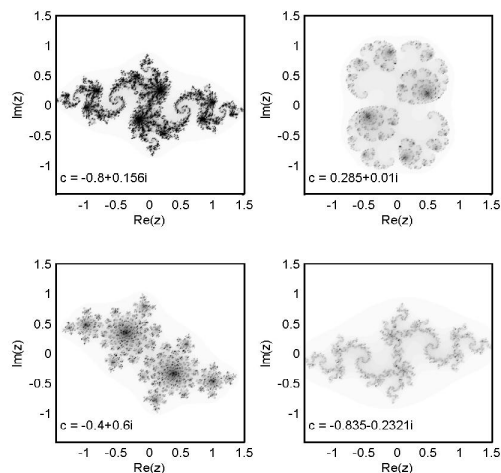
$$\begin{aligned} z_0 &= p \\ z_{n+1} &= z_n^2 + c \end{aligned} \quad (6)$$

W zależności od punktu startu p możemy w wyniku iteracji otrzymać ciąg, który będzie nieograniczony (jego elementy opuszczają każdy okrąg ze środkiem w centrum układu współrzędnych) lub ograniczony (czyli taki, dla którego istnieje okrąg o środku w centrum układu współrzędnych, jakiego elementy ciągu nigdy nie opuszczają). Zatem zgodnie z definicją (1) zbiór wypełniony Julii tworzą te punkty p , dla których ciąg jest ograniczony, w szczególności można dowiedzieć, że punkty takie spełniają warunek $|z_n| < 2$ dla $n \rightarrow \infty$. Formalnie obraz takiego zbioru powinien być czarno-biały, gdzie piksele, np. koloru białego, odpowiadałyby punktom spełniającym wyżej wymieniony warunek. W praktyce jednak liczba iteracji jest ograniczana, a kolory pikseli odpowiadające punktom p ustala się w sposób arbitralny na podstawie osiągniętej liczby iteracji (czyli szybkości, z jaką ciąg zbiega do nieskończoności w danym punkcie p). W ten sposób powstają barwne obrazy, potocznie zwane fraktalami. Rysunek 1 przedstawia przykłady popularnych obrazów fraktalnych z rodziny zbiorów Julii, uzyskanych dla wybranych parametrów c . Poniżej zamieszczono też pseudokod typowego algorytmu generowania obrazów zbiorów Julii na płaszczyźnie zespolonej.

```

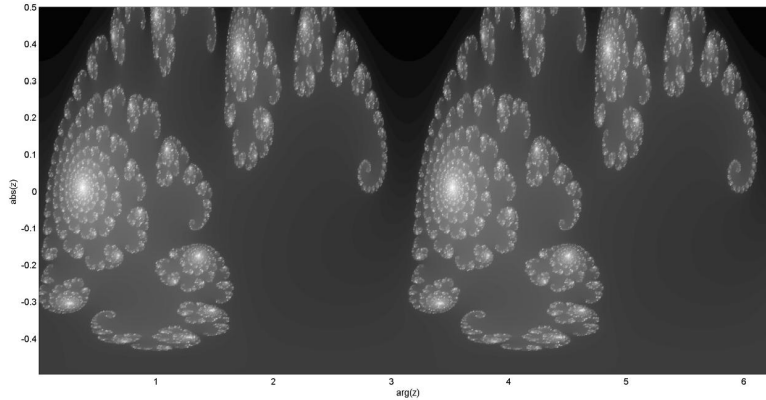
for ui = 1 to width do
  for vi = 1 to height do
    z_n = 3*(ui/width - 0.5) + 1i*3*(vi/height - 0.5)
    n = 0
    while abs(z_n) < 2 and n < n_max do
      z_n = z_n^2 + c
      n = n + 1
    end
    texture[ui][vi] = color[n]
  end
end
end

```



Rys. 1. Obrazy zbiorów Julii na płaszczyźnie zespolonej dla różnych wartości parametru c

Fig. 1. Pictures of the Julia sets on the complex plane for different values of parameter c



Rys. 2. Zbiór Julii otrzymany dla interpretacji biegunowej współrzędnych tekstur
 Fig. 2. Julia set obtained for polar interpretation of the texture coordinates

Oznaczmy przez u, v rzeczywiste, znormalizowane współrzędne tekstury, takie że $t(u=0, v=0)$ to piksel w lewym dolnym rogu tekstury oraz $t(u=1, v=1)$ to piksel w prawym górnym rogu. Wówczas punkty startowe iteracji wielomianu (6) mogą być obliczane zgodnie ze wzorem:

$$p = s_u(u-0.5) + i \cdot s_v(v-0.5) , \quad (7)$$

gdzie s_u, s_v są to rzeczywiste współczynniki skalujące, określające rozmiar prostokątnego obszaru próbkowania płaszczyzny zespolonej odpowiednio w osi rzeczywistej i urojonej. Częściej jednak tekstury nieba odwzorowywane są na półsferę, z wykorzystaniem rzutowania cylindrycznego lub sferycznego. Tekstury takie powinny posiadać dopasowane krawędzie boczne, tj. $t(0, v) = t(1, v)$ dla dowolnego v . Dlatego też bardziej korzystne wydaje się próbkowanie płaszczyzny zespolonej w układzie współrzędnych biegunowych. Wówczas zmienne u, v interpretujemy jako znormalizowane współrzędne biegunowe, a punkty startowe obliczamy zgodnie ze wzorem:

$$p = r \cdot v \exp(i2\pi u) , \quad (8)$$

gdzie r – rzeczywisty współczynnik skalujący. Zakładając równomierne rozmieszczenie próbek (u, v) w układzie prostokątnym tekstury, uzyskujemy próbki p rozmieszczone równomiernie na okręgach (ze środkiem w punkcie $(0, 0)$ i promieniach rv), na płaszczyźnie zespolonej. Rysunek 2 przedstawia uzyskaną w ten sposób teksturę. Reprezentuje ona wewnątrz okręgu jednostkowego (dla $r=1$) zbioru z rysunku 1 (prawy górny róg).

3 Spaczone zbiory Julii

Jak można zaobserwować na rysunku 1, struktury fraktalne zbiorów Julii, mają nietrywialny charakter, wykazują jednak pewne symetrie względem środka układu współrzędnych lub w przypadku reprezentacji biegunowej – okresowość (rys. 2). W zastosowaniach związanych z symulacją nieba jest to raczej cecha niepożądana,

która uniemożliwia bezpośrednie wykorzystanie obrazów zbiorów Julii jako tekstur. Dlatego też w niniejszej pracy poszukujemy takiej metody deformacji struktur fraktalnych, która pozwoliłaby na usunięcie nienaturalnie wyglądających symetrii/okresowości przy jednoczesnym zachowaniu nietrywialnej struktury zbioru. Tematyka związana ze zniekształcaniem fraktali jest poruszana w pracy [6], jednak dotyczy ona przypadku metamorfozy. Proponowane podejście jest dużo prostsze i czerpie z teorii cyfrowych filtrów wszechprzepustowych. Teoria ta znajduje zastosowanie głównie przy projektowaniu spaczonych banków filtrów oraz transformatach blokowych [7]. Filtry wszechprzepustowe wykorzystuje się między innymi do deformacji (tzw. spaczania) osi częstotliwości (ang. *frequency warping*). Transmitancja filtru pierwszego rzędu definiowana jest następująco [8]:

$$A(z) = \frac{z^{-1} - \alpha^*}{1 - \alpha z^{-1}}, \quad (9)$$

gdzie α – parametr zespolony, określający jednocześnie biegun filtru (tj. $A(z = \alpha) \rightarrow \infty$). Filtr taki jest stabilny wtedy i tylko wtedy, gdy $|\alpha| < 1$.

W przypadku spaczania osi częstotliwości przyjmuje się, że wartości z położone są na okręgu jednostkowym, wówczas funkcja (9) nie modyfikuje modułu liczby zespolonej, a jedynie jej fazę. Pozwala to na zagęszczanie próbek częstotliwościowych na okręgu jednostkowym wokół wybranego punktu $e^{-i\arg(\alpha)}$, proporcjonalnie do wartości modułu $|\alpha|$. Natomiast w ogólnym przypadku zachodzą zależności:

$$|A(z)| = \begin{cases} > 1, & |z| < 1 \\ = 1, & |z| = 1 \\ < 1, & |z| > 1 \end{cases}. \quad (10)$$

Istotą prezentowanej metody jest deformowanie/spaczanie regularnej siatki próbek startowych na całej płaszczyźnie zespolonej, a nie tylko na okręgu jednostkowym. Mianowicie proponujemy następującą procedurę iteracyjną:

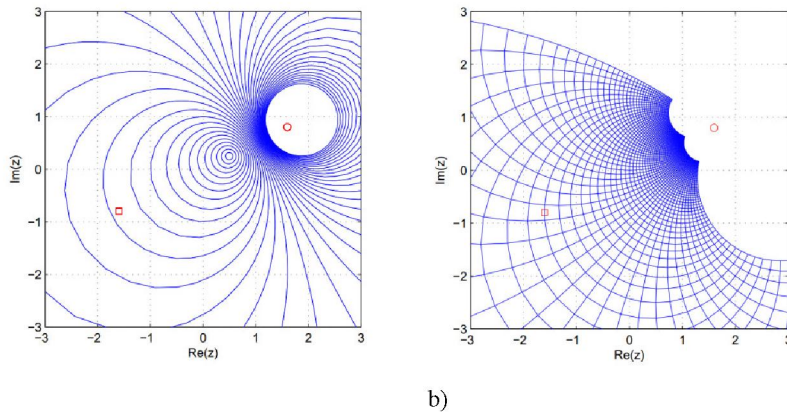
$$\begin{aligned} \hat{z}_0 &= A(p^{-1}) \\ \hat{z}_{n+1} &= \hat{z}_n^2 + c \end{aligned}. \quad (11)$$

Z tej perspektywy kwestie związane ze stabilnością filtru nie są istotne i parametr α może być dowolną liczbą rzeczywistą lub zespoloną. Interesuje nas raczej jego wpływ na wygląd spaczonej siatki próbek startowych, a co za tym idzie, interpretacja geometryczna przekształcenia wszechprzepustowego. Zauważmy, że wyrażenie (9) stanowi w istocie szczególny przypadek transformacji Möbiusa:

$$g(z) = \frac{az + b}{cz + d}, \quad (12)$$

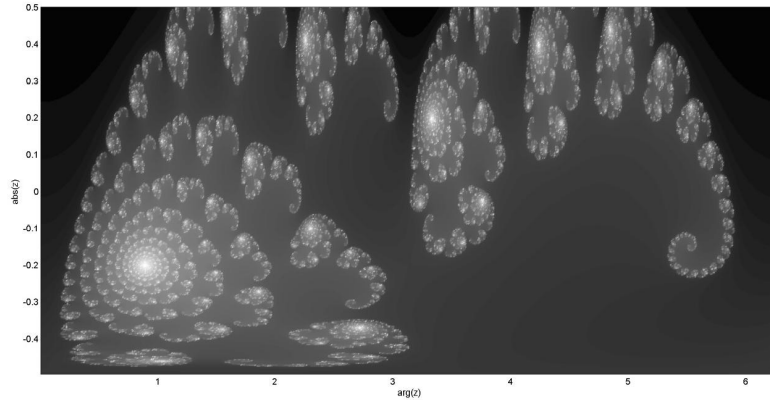
gdzie a, b, c, d to liczby zespolone spełniające warunek $ad - bc \neq 0$. Jeśli przyjmiemy, że $a = d = 1, b = -\alpha^*$ oraz $c = -\alpha$, wówczas $A(z^{-1}) = g(z)$. Transformację Möbiusa można interpretować na płaszczyźnie zespolonej jako złożenie kilku przekształceń, a mianowicie: rzutowania stereograficznego punktów płaszczyzny na sferę, następnie

jej przesunięcia, obrotu, dylatacji i w końcu przekształcenia odwrotnego, rzutującego punkty ze sfery na płaszczyznę. Transformacje takie zachowują kąty, odwzorowując każdą prostą na prostą lub okrąg oraz każdy okrąg na okrąg lub prostą. Posiadają też kilka punktów charakterystycznych, m.in. biegun $z_{\infty} = -d/c = 1/a$, czyli taki punkt, dla którego $g(z_{\infty}) \rightarrow \infty$, i antybiegun $Z_{\infty} = a/c = -1/a$, dla którego $g(\infty) \rightarrow Z_{\infty}$. Dokładniej ilustruje to rysunek 3, przedstawiający obrazy transformacji punktów startowych rozmieszczonych regularnie na okręgach oraz na siatce prostokątnej (dla $a = 0.5 + 0.25i$), kwadratem i kółkiem oznaczono odpowiednio biegun i antybiegun. W miejscach, gdzie węzły siatki są zagęszczane, odpowiadające im struktury fraktalne będą powiększane, natomiast w miejscach, gdzie węzły są rozciągane, odpowiednie struktury będą pomniejszane. Ponieważ części struktur są pomniejszoną kopią całości (samopodobieństwo) uzyskany efekt może przypominać „zapadanie się” fraktala. Przypadek taki dobrze ilustruje rysunek 4.



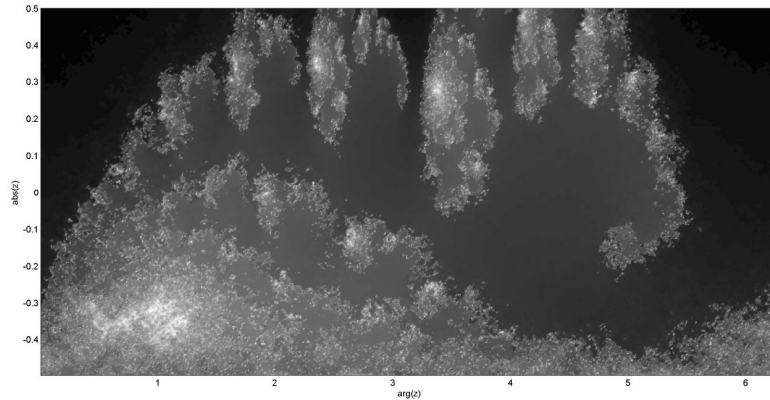
Rys. 3. Obrazy transformacji wszechprzepustowej punktów równomiernie rozmieszczonych na okręgach (a) i siatce prostokątnej (b) ze środkiem w punkcie $(0,0)$

Fig. 3. Images of allpass-transformed equidistant points placed on circles (a) and rectangular grid (b) centered on the point $(0,0)$



Rys. 4. Spaczona wersja zbioru Julii z rysunku 2, otrzymana przy użyciu transmitancji filtra wszechprzepustowego z parametrem $\alpha = -0.3$

Fig. 4. Warped version of the Julia set of the figure 2, obtained using the transfer function of the allpass filter with parameter $\alpha = -0.3$



Rys. 5. Spaczony zbiór Julii po dodaniu turbulencji o amplitudzie $\lambda = 0.2$

Fig. 5. Warped Julia set after adding some turbulence with amplitude $\lambda = 0.2$

Dalsze zmniejszanie wartości parametru α powoduje przesuwanie się i jednoczesną kompresję struktur spiralnych w lewo i ostateczne „zapadnięcie się” fraktala w sobie (ze względu na ograniczenie objętościowe nie zamieszczono odpowiednich rysunków).

Spaczenie struktur fraktalnych z reguły pozbawia je symetrii, jednak zachowują one cechy samopodobieństwa. W niektórych zastosowaniach może okazać się to wystarczające, jednak w przypadku proponowanej metody wprowadzono dodatkowo efekt turbulencji poprzez proste zaburzenie spaczonej siatki punktów startowych zespolonym szumem proceduralnym $W(p) \in \langle 0;1 \rangle$ o niewielkiej amplitudzie, tj.:

$$\hat{z}_0 = A(p^{-1}) + \lambda W(p), \quad (13)$$

gdzie parametr λ określa amplitudę. W trakcie eksperymentów do symulacji szumu $W(p)$ wykorzystano tzw. ułamkowe ruchy Browna (ang. *fractional Brownian motion - fBm*) [3]. Szum fBm jest powszechnie wykorzystywany w proceduralnym generowaniu terenu i chmur, posiada też cechy samopodobieństwa stochastycznego, jest więc fraktalem. Uzyskany efekt przedstawiono na rysunku 5.

4 Wnioski

Przedstawiono nową metodę symulacji tekstur w oparciu o spaczony zbiory Julii. Zbiory te otrzymano w wyniku spaczenia (deformacji) siatki punktów startowych dla procesu iteracji funkcji wymiernej, wykorzystując transmitancję filtru wszechprzepustowego pierwszego rzędu. Wyniki eksperymentów pokazują, że proponowana metoda pozwala usunąć nienaturalnie wyglądające symetrie lub okresowość w obrazach zbiorów Julii, dzięki czemu po dodaniu niewielkich turbulencji można wykorzystać je bezpośrednio do symulacji tekstur nieba nocnego. Metoda ta daje relatywnie dobre rezultaty, jednocześnie jest prosta w implementacji. W niniejszej pracy wszystkie symulacje przeprowadzono w środowisku Matlab, jednak przyszłe prace obejmują implementacje sprzętowe w programach typu *pixel-shader*. W obszarze naszych zainteresowań jest również rozszerzenie idei spaczania obrazów fraktalnych do liczb hiperzespólnych, a także wykorzystanie transmitancji filtrów wyższych rzędów.

Literatura

1. Xiaoyan Hu, Sun Bo, Ren Wei, Liang Xiaohui, Wu YunFeng: Cloud System for Flight Simulation, Information Science and Engineering (ICISE), 2009 *1st International Conference*, pp. 1196-1199, 26-28 Dec. 2009
2. Roden T., Parberry I.: Clouds and stars: efficient real-time procedural sky rendering using 3D hardware, *Proc. of the 2005 ACM SIGCHI Int. Conf. on Advances in Computer Entertainment Technology*, Valencia, Spain, pp. 434-437, 2005
3. Ebert D.S., Musgrave F.K., Peachey D., Perlin K., Worley S.: *Texturing and Modeling: A Procedural Approach (3rd ed.)*. Morgan Kaufmann Inc., 2002
4. Kudrewicz J.: *Fraktale i chaos*, Wyd. 3. WNT, Warszawa 1996
5. Falconer K.J., *Fractal Geometry: Mathematical Foundations and Applications 2nd ed.* Chichester, West Sussex: John Wiley and Sons, 2003
6. Martyn T., A new approach to morphing 2D affine IFS fractals, *Computers & Graphics*, vol. 28, no. 2, pp. 249-272, April 2004
7. Borowicz A., Parfieniuk M., Petrovsky A.A.: An application of the warped discrete Fourier transform in the perceptual speech enhancement, *Speech Communication*, vol. 48, no. 8, pp. 1024-1036, August 2006
8. Regalia P.A., Vaidyanathan P.P., Mitra S.K.: The digital allpass filter: a versatile signal processing building block, *Proc. IEEE*, vol. 76, no. 1, pp. 19-37, Jan 1988

Streszczenie

Przedstawiono nową technikę symulacji tekstur nieba nocnego na potrzeby systemów rzeczywistości wirtualnej. Proponowana metoda wykorzystuje w sposób bezpośredni spaczone zbiory Julii. W odróżnieniu od klasycznej metody wykorzystuje się tu przekształcenie wszechprzepustowe do spaczania (deformacji) siatki punktów startowych. Dzięki temu możliwe jest uzyskanie bardziej realistycznych struktur fraktalnych, pozbawionych nienaturalnych symetrii lub okresowości, zachowujących jednak cechy samopodobieństwa.

Słowa kluczowe: symulacja tekstur, zbiór Julii, filtr wszechprzepustowy

An application of the warped Julia sets in the sky texture simulation

Summary

The paper presents a new technique to simulate night sky textures for the purposes of virtual reality systems, in particular computer games. The proposed method uses the warped Julia sets directly. In contrast to the classical method, here an allpass transformation is used first to warp (deform) a grid of the starting points. This makes it possible to obtain more realistic fractal structures, without unnatural symmetry or periodicity, but preserving the self-similarity features.

Keywords: texture simulation, Julia set, allpass filter

Praca finansowana w ramach badań statutowych Wydziału Informatyki Politechniki Białostockiej nr S/WI/1/2013.