

Łukasz Jopek\*, Robert Nowotniak\*, Michał Postolski\*,  
Laurent Babout\*, Marcin Janaszewski\*

## Zastosowanie kwantowych algorytmów genetycznych do selekcji cech

### 1. Wprowadzenie

Niniejszy artykuł dotyczy problemu automatycznej selekcji cech, wykorzystywanych przez klasyfikatory w procesie segmentacji obrazów. Dokładna segmentacja jest istotnym krokiem w wielu praktycznych zastosowaniach polegających na analizowaniu obrazów, np. analiza ilościowa i jakościowa obrazów pochodzących z tomografii komputerowej lub rezonansu magnetycznego w materiałoznawstwie [7, 9, 10, 11] lub medycynie [14]. Od jej wyniku zależy, czy kolejne etapy całego procesu wykonają się prawidłowo, a uzyskane wyniki, np. analizy ilościowej, będą poprawne.

Selekcja cech może zostać potraktowana jako problem dwukryterialnej optymalizacji kombinatorycznej, której celem jest znalezienie takiego podzbioru cech opisujących punkty obrazu, który umożliwi uzyskanie zadowalającej jakości segmentacji w możliwie krótkim czasie. W artykule została porównana skuteczność trzech populacyjnych metod heurystycznych: klasycznego algorytmu genetycznego oraz jego dwóch modyfikacji, czerpiących inspirację z systemów informatyki kwantowej: QiGA (*Quantum-Inspired Genetic Algorithm*) oraz GAQPR (*Genetic Algorithm with Quantum Probability Representation*).

Treść artykułu została podzielona w następujący sposób: w części drugiej artykułu zostało umieszczone wprowadzenie w zagadnienie segmentacji obrazów oraz w istotę problemu selekcji cech dla klasyfikatorów. W części trzeciej zagadnienie selekcji cech zostało sformułowane w postaci problemu optymalizacji kombinatorycznej. W części czwartej zostały przedstawione trzy warianty algorytmu genetycznego, których skuteczność została porównana. Część piąta zawiera wyniki, przeprowadzonego porównania dla tak postawionego problemu. Część szósta podsumowuje uzyskane przez autorów rezultaty oraz wyznacza możliwe kierunki dalszych badań.

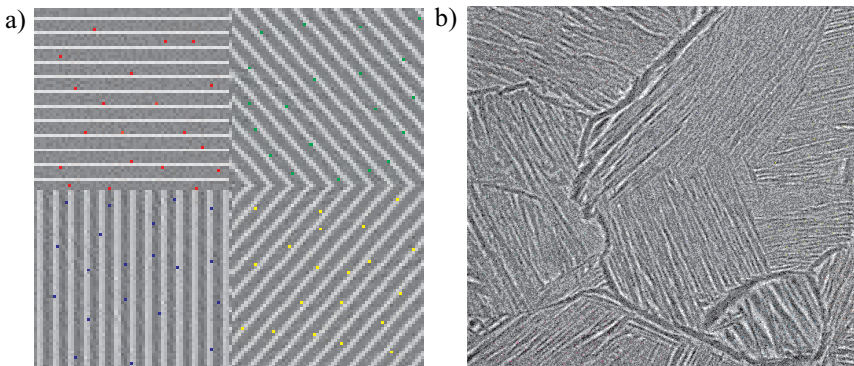
---

\* Katedra Informatyki Stosowanej, Politechnika Łódzka

## 2. Segmentacja obrazów

Segmentacja obrazu to podział obrazu na obszary, jednorodne pod względem wybranych kryteriów. W niniejszym artykule segmentacja rozważana jest jako przypisywanie kolejnych punktów obrazu do odpowiednich klas z wcześniej zdefiniowanego zbioru. W tym celu został wykorzystany klasyfikator minimalnoodległościowy  $k$ -NN, działający w przestrzeni cech punktów obrazu.

W procesie segmentacji korzysta się zazwyczaj z różnorodnych cech punktów obrazu. Liczba możliwych do wyznaczenia cech dla obrazów teksturowanych (rys. 1) wynosi obecnie około 300 i tego rodzaju analiza może być wykonana za pomocą oprogramowania takiego jak program MaZda [6]. Do przeprowadzenia poprawnej segmentacji jest jednak ważne, by wybrać tylko te cechy lub grupy cech, które są dobrymi dyskryminatorami, czyli cechami, które pozwalają jak najlepiej rozróżnić dane klasy obrazów. Ich liczbowe deskryptory, zwane dyskryminatorami klas, powinny wykazywać możliwie dużą różnicę między przedstawicielami różnych klas, a możliwie małe różnice dla elementów wewnątrz danej klasy. W pewnych przypadkach można przeprowadzić segmentację, wykorzystując tylko jedną cechę, ale często jest to zespół kilku cech.



**Rys. 1.** Obraz tekstury sztucznej (a) oraz tomograficznej [10, 11] (b), na których zostały przeprowadzone badania. Wyróżnione punkty na obrazie (a) są losowo wybranymi reprezentantami każdej z klas i stanowią będą zbiór uczący w procesie klasyfikacji

Najprostszą metodą wyboru cech jest wykorzystanie współczynnika Fishera [9], wyznaczanego dla każdej z cech osobno i wybór tej cechy, dla której współczynnik ten osiąga wartość najwyższą. Algorytm sprawdza się wtedy, gdy do segmentacji wystarczy jedna cecha, natomiast jest nieskuteczny, gdy optymalnym okaże się zbiór kilku cech. Tutaj z pomocą przychodzą takie algorytmy jak algorytm sekwencyjnej selekcji wstecznej SBS [8] (*Sequential Backward Selection*), algorytm sekwencyjnej selekcji postępującej SFS [8]

(*Sequential Forward Selection*), czy powstały z połączenia tych metod algorytm FSM [8] (*Floating Search Methods*). Inne możliwe metody selekcji cech mogą wykorzystywać algorytm PCA (*Principal Component Analysis*), MDS [7] (*Multidimensional Scaling*) lub POE+ACC [7] (*Probability of Error + Average Correlation Coefficient*).

Żadna z wyżej wymienionych metod nie gwarantuje jednak znalezienia zbioru cech, posiadającego największą siłę dyskryminacji. Jediną metodą, która mogłaby to zapewnić, byłoby sprawdzenie wszystkich możliwych podzbiorów zbioru cech. Innym kierunkiem może być wyznaczenie najkorzystniejszego w sensie określonej funkcji jakości podzbioru na drodze optymalizacji. Przedmiotem niniejszego artykułu jest zbadanie efektywności wybranych metod optymalizacji ewolucyjnej w zaprezentowanym problemie selekcji cech.

### 3. Problem selekcji cech

Podział obrazu na obszary poprzez klasyfikację punktów może być wykonany na podstawie wielu różnych cech punktów obrazu [7, 12]. Na potrzeby niniejszego artykułu zostały wybrane następujące grupy cech:

- Cechy statystyczne histogramu (6 cech).
- Cechy wyznaczone na podstawie macierzy zdarzeń (120 cech).
- Cechy wyznaczone na podstawie pasm (16 cech).
- Cechy wyznaczone na podstawie macierzy gradientu (2 cechy).

Co łącznie daje 144 różne cechy. Dla każdego piksela zostały one obliczone przy wykorzystaniu okna o rozmiarach  $25 \times 25$  pikseli, w którego centrum znajdował się badany piksel.

Warto zauważyć, że wykorzystywanie niektórych cech wiąże się ze znacznym kosztem obliczeniowym, a ich wpływ na poprawność klasyfikacji punktów (a zatem na jakość segmentacji) jest niewielki. W skrajnym przypadku wykorzystanie niektórych cech może nawet powodować zwiększenie kosztu obliczeniowego segmentacji przy jednoczesnym pogorszeniu jej jakości.

Z tego powodu wybór podzbioru cech, które będą wykorzystywane przez klasyfikator, można traktować jako zadanie optymalizacji kombinatorycznej, uwzględniającej dwa kryteria: czas potrzebny na obliczenie wartości wybranych cech oraz uzyskiwaną jakość segmentacji.

Ponieważ każdy  $n$ -elementowy zbiór posiada  $2^n - 1$  niepustych podzbiorów, liczba rozwiązań, które należałoby sprawdzić, poszukując optymalnego podzbioru cech, rośnie w tempie wykładniczym wraz ze wzrostem liczby uwzględnianych cech.

Przy założeniu, że pojedynczy proces klasyfikacji za pomocą algorytmu  $k$ -NN trwa 10 ms, ocena wszystkich podzbiorów 32-elementowego zbioru cech, wymagałaby ponad roku obliczeń. Z tego powodu rozsądną alternatywą jest podejście heurystyczne, pozwalające

jące na efektywne znalezienie satysfakcjonującego, suboptymalnego rozwiązania w znacznie krótszym czasie.

Przestrzenią rozwiązań dla analizowanego problemu jest przestrzeń wszystkich podzbiorów uwzględnianych cech. Naturalną reprezentacją pojedynczego podzbioru jest wektor binarny (lub odpowiadający mu ciąg zero-jedynkowy) o długości równej liczbie wszystkich cech. Wartości (0, 1) poszczególnych elementów ciągu określają przynależność cechy do danego podzbioru. Na podstawie takiego wektora może być budowany zbiór uczący i zbiór testujący dla klasyfikatora  $k$ -NN. Wskaźnikiem jakości danego zbioru cech jest uzyskany rezultat klasyfikacji. Im większy jest procent poprawnie zaklasyfikowanych obiektów, tym wybrany zbiór cech lepiej dyskryminuje dany obraz.

Na potrzeby niniejszego artykułu funkcja oceny dla problemu selekcji cech została dobrana w następujący sposób:

$$f(x) = K - S(t, w) \quad (1)$$

gdzie:

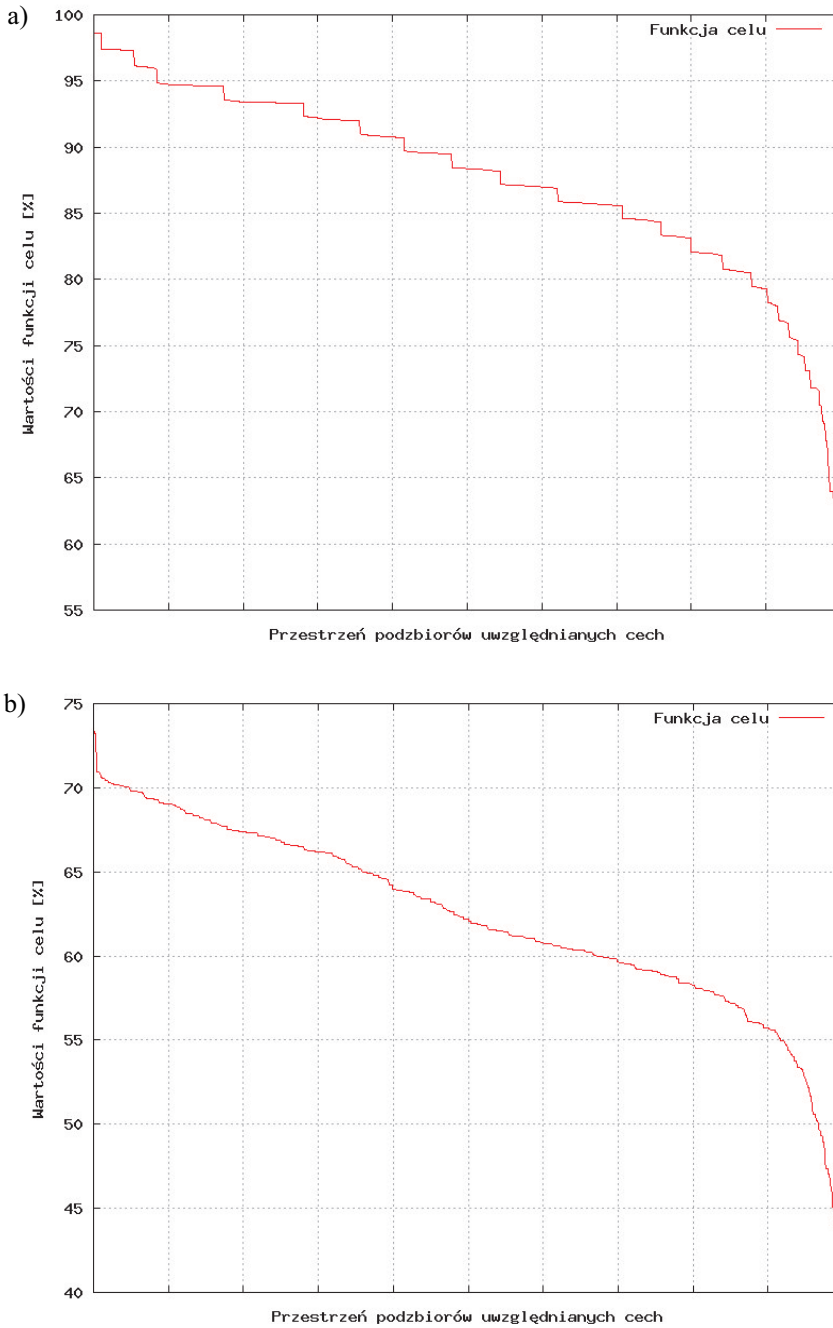
- $x$  – wybrany podzbiór wszystkich uwzględnianych cech,
- $K$  – jakość uzyskiwanej klasyfikacji,
- $t$  – czas potrzebny na obliczenie cech, należących do  $x$ ,
- $S(t, w)$  – liniowa funkcja kary, skalująca czas  $t$  do przedziału  $\langle 0; w \rangle$ .

W celu ostatecznej oceny poprawności segmentacji testowe obrazy (rys. 1) zostały wstępnie podzielone przez człowieka na obszary. Jakość klasyfikacji  $K$  została określona poprzez procentową miarę poprawnie zaklasyfikowanych punktów przez klasyfikator  $k$ -NN działający na zbiorze testowym. Dlatego dziedziną funkcji oceny jest przedział  $\langle 0; 100\% \rangle$ .

Funkcja  $S(t, w)$  służy do przeskalowania czasu obliczania wartości cech  $t$  do przedziału  $\langle 0; w \rangle$ . W artykule przyjęto, że wpływ czynnika czasu  $w = 10\%$ , co oznacza, że wybranie wszystkich cech pogorszy wartość funkcji oceny o 10%. Zakres wartości dla funkcji celu  $f(x)$ , w tym przypadku, nie wynosi  $\langle 0; 100\% \rangle$ , ponieważ wartość maksymalna ograniczona jest przez wpływ funkcji kary  $S(t, w)$ . Jeśli  $S(t, w) = 0$ , to zakres wartości dla funkcji celu  $f(x)$  wynosi  $\langle 0; 100\% \rangle$ .

Należy zauważyć, że przy zerowym wpływie czynnika czasu (brak penalizacji rozwiązań ze względu na czas), wartość funkcji celu może być bezpośrednio interpretowana jako część punktów ze zbioru testującego, która została w prawidłowy sposób przydzielona do klas.

Rysunek 2 przedstawia rozkład statystyczny wartości funkcji celu w przestrzeni podzbiorów zbioru dziesięciu, wybranych cech. Przedstawione wykresy pozwalają odczytać m.in. jaką część spośród wszystkich podzbiorów cech pozwala na uzyskanie wartości funkcji oceny powyżej pewnego poziomu.



**Rys. 2.** Rozkład statystyczny wartości zaproponowanej przez autorów funkcji celu w przestrzeni podzbiorów uwzględnianych cech – dla tekstury sztucznej (a) oraz tomograficznej (b)

## 4. Kwantowe algorytmy genetyczne

Informatyka kwantowa [15] zajmuje się wykorzystaniem możliwości obliczeniowych miniaturowych układów, podlegających prawom mechaniki kwantowej. Dzięki unikalnym własnościom takich układów niektóre problemy obliczeniowe mogłyby być rozwiązywane w sposób znacznie bardziej efektywny, niż jest to możliwe za pomocą jakichkolwiek klasycznych maszyn obliczeniowych (np. [4]). Jedną z największych przeszkód w budowie komputera kwantowego są jednak obecnie problemy z utrzymywaniem stanu koherencji kwantowej, pojawiające się wraz ze wzrostem rozmiarów rejestrów kwantowych.

Mimo iż budowa użytecznego komputera kwantowego pozostaje ciągle wyzwaniem, równoległe od późnych lat 90. XX wieku trwają badania nad metodami sztucznej inteligencji, posiadającymi elementy *inspirowane* systemami informatyki kwantowej. W ciągu ostatniego dziesięciolecia wykazano, że modyfikacja istniejących metod sztucznej inteligencji, przez dodanie elementów o naturze kwantowej, pozwala w wielu przypadkach na znaczną poprawę efektywności tych metod. Tego typu metody hybrydowe są na świecie przedmiotem badań w chwili obecnej [5].

Jedną z prób wprowadzenia tego typu modyfikacji do algorytmów genetycznych są algorytmy QiGA [1, 2] (*Quantum-Inspired Genetic Algorithm*) oraz GAQPR [3] (*Genetic Algorithm with Quantum Probability Representation*). Dla uproszczenia zapisu te modyfikacje algorytmów będą w dalszej części artykułu nazywane kwantowymi algorytmami genetycznymi. Mimo iż z założenia realizacja tych algorytmów nie wymaga wykorzystania maszyn obliczeniowych działających na poziomie kwantowym, zasada ich działania nie wyklucza możliwości ich implementacji w strukturze komputera kwantowego.

Cechą charakterystyczną wymienionych kwantowych algorytmów genetycznych jest reprezentacja rozwiązań, wykorzystująca koncepcję *kubitów*, będących podstawowymi jednostkami informacji w informatyce kwantowej. Zmodyfikowane operatory genetyczne, używane do iteracyjnego ulepszania populacji rozwiązań, są natomiast wzorowane na ewolucji unitarnej układu kwantowego. Nowy, kwantowy operator genetyczny odpowiada operacji obrotu wektora stanu w przestrzeni stanów kubitów, opisującego stan genu w chromosomie kwantowym. Wprowadzenie tego dodatkowego elementu losowości do algorytmu genetycznego pozwala na poprawę efektywności rozwiązywania niektórych problemów kombinatorycznych, opisywanych przez nieciągłe, silnie nieliniowe funkcje celu [1].

Zasadę działania algorytmu GAQPR przedstawia poniższy pseudokod:

**procedure** GAQPR

**begin**

$t \leftarrow 0$

  zainicjalizuj  $Q(t)$

  utwórz  $P(t)$  poprzez obserwację stanów  $Q(t)$

  ocień rozwiązanie  $P(t)$

  zapamiętaj najlepsze rozwiązanie z  $P(t)$

**while** (warunek stopu nie spełniony) **do**

**begin**

$t \leftarrow t + 1$

utwórz  $P(t)$  poprzez obserwację stanów  $Q(t-1)$

ocień rozwiązanie  $P(t)$

zapamiętaj najlepsze rozwiązanie z  $P(t)$

zaktualizuj  $Q(t)$  za pomocą bramek kwantowych  $U(t)$

*zastosuj dodatkowe operatory genetyczne na  $Q(t)$*

**end**

**end**

gdzie:

$t$  – numer generacji,

$Q(t)$  – populacja chromosomów kwantowych w generacji  $t$ ,

$P(t)$  – zbiór ciągów binarnych, za pomocą którego dokonywana jest ocena rozwiązań,

$U(t)$  – zbiór bramek kwantowych używanych w generacji  $t$  do modyfikacji populacji  $Q$ .

GAQPR jest w istocie rozszerzeniem algorytmu QiGA o nowe operatory genetyczne (ostatni etap iteracji), mające na celu dodatkowe wzbogacenie różnorodności genetycznej populacji rozwiązań i tym samym przeciwdziałanie przedwczesnej zbieżności algorytmu. Przedstawiony pseudokod opisuje zatem również algorytm QiGA, jeśli pominiemy te dodatkowe parametry.

## 5. Uzyskane wyniki

Wymienione w poprzedniej części kwantowe algorytmy genetyczne zostały zastosowane przez autorów artykułu do rozwiązania zadania optymalizacyjnego, przedstawionego w części trzeciej, a uzyskane wyniki porównano z rozwiązaniami znalezionymi za pomocą prostego algorytmu genetycznego. W tej części artykułu zostały przedstawione warunki przeprowadzonych testów, uzyskane wyniki oraz końcowy efekt segmentacji.

Wybrane metody ewolucyjne przetestowano na dwóch typach obrazów teksturowanych (rys. 1). Pierwszym był sztucznie wygenerowany obraz (1a), a drugim była rzeczywista tekstura (1b), przedstawiająca powierzchnię krystalicznego stopu tytanu, wykonana techniką mikrotomografii rentgenowskiej. Punkty tych dwóch obrazów zostały podzielone odpowiednio na 4 i 8 klas. Na każdym z obrazów wyznaczono manualnie obszary poszczególnych klas, a następnie z każdej klasy wybrano pewną liczbę pikseli, które stały się obiektami uczącymi dla klasyfikatora  $k$ -NN. Liczba była zależna od rodzaju obrazu. Natomiast liczba obiektów w poszczególnych klasach była taka sama lub zbliżona do siebie.

Obraz pierwszego typu (tekstura sztuczna) posłużył do testowania poprawności działania algorytmu, ponieważ dla tego obrazu łatwo jest wyznaczyć optymalny zbiór cech. Trudniejszym zadaniem była segmentacja obrazu tekstury tomograficznej. Pomimo że

wcześniejsze analizy wykazały, że decydującą rolę powinny odgrywać cechy badające kierunkowość tekstury, znalezienie optymalnego zbioru okazało się trudne.

W każdej klasie w sposób losowy wybrano 20 punktów, które utworzyły zbiór uczący i testujący dla klasyfikatora  $k$ -NN. Wybór jednakowych punktów dla zbioru uczącego i testującego może sztucznie podnosić jakość uzyskanych wyników, jednakże głównym celem artykułu było porównanie skuteczności kwantowych algorytmów genetycznych z algorytmami klasycznymi.

W zastosowanym prostym algorytmie genetycznym przyjęto typowe dla takiego algorytmu parametry. Zastosowano krosowanie i mutacje jednopunktowe, z prawdopodobieństwami odpowiednio  $P_c = 95\%$  i  $P_m = 5\%$ . Na etapie selekcji użyto metody turniejowej. Dla algorytmu GAQPR przyjęto parametry  $P_c=10\%$  i  $P_m=10\%$  i kąty obrotów w przestrzeni stanów kubitu odpowiednio  $4^\circ$  i  $8^\circ$ . Warunkiem stopu procesu optymalizacji było w każdym przypadku wykonanie ustalonej z góry liczby iteracji (20). Dla każdego z trzech porównanych algorytmów przyjęto licznosc populacji wynoszącą 50 osobników. W klasyfikatorze  $k$ -NN zastosowano metrykę euklidesową oraz przyjęto liczbę najbliższych sąsiadów wynoszącą 5.

Dla zwiększenia wiarygodności otrzymanych wyników dla każdego algorytmu i typu tekstury proces optymalizacji był wykonywany dziesięciokrotnie, a wykresy przedstawiają średnie wyniki uzyskane podczas dziesięciu testów.

Porównywane algorytmy zostały zaimplementowane przez autorów artykułu w języku Java, a testy zostały przeprowadzone na komputerze klasy PC z procesorem AMD Sempron 2800. Otrzymane wyniki przedstawione są w tabeli 1.

Każdy zbiór cech, zawierający 19, 32, 48 czy 144 elementy, został dobrany tak, aby posiadał cechy potencjalnie dobre oraz potencjalnie nie przydatne. Informacja o przydatności danej cechy pochodziła z wcześniejszych analiz badanych obrazów. Zbiór bardziej liczny jest potencjalnie lepszy, bowiem pozwala sprawdzić więcej kombinacji cech, jednak zwiększenie przestrzeni rozwiązań pociąga za sobą znaczne wydłużenie czasu pracy algorytmu. Dlatego w zastosowaniach praktycznych, bazując na doświadczeniu i wiedzy na temat obrazów, które poddane będą segmentacji, wstępnie dokonuje się ograniczenia zbioru cech. Dokonane przez autorów pomiary dla różnych długości wektora cech miały na celu sprawdzenie, czy wstępna selekcja jest konieczna.

Jak łatwo zauważyć, tekstura wygenerowana sztucznie, zgodnie z przewidywaniami, uzyskała najlepszy możliwy wynik, czyli średnio 98% poprawnie zaklasyfikowanych obiektów. Wynik 100% nie był możliwy do osiągnięcia ze względu na zastosowaną karę zależną od czasu. Ponadto w przypadku tej tekstury ilość cech, z których dokonywana jest selekcja, nie wpływał znacząco na wynik. Jest to spowodowane bardzo małym skomplikowaniem tak przygotowanej tekstury, co sprawia, że nawet pojedyncze cechy dają bardzo wysoką jakość klasyfikacji.

W przypadku tekstury tomograficznej wyniki były już gorsze, jest to jednak zgodne z przewidywaniami. Klasy w zbiorze uczącym nie są rozdzielne liniowo, co zostało sprawdzone odpowiednim algorytmem [13]. Jednakże celem algorytmu jest znalezienie jak naj-



lepszego zbioru cech, co w praktyce nie zawsze oznacza poprawną klasyfikację w każdym przypadku i pewny, niewielki procent obiektów (pikseli obrazu) może być klasyfikowany niepoprawnie.

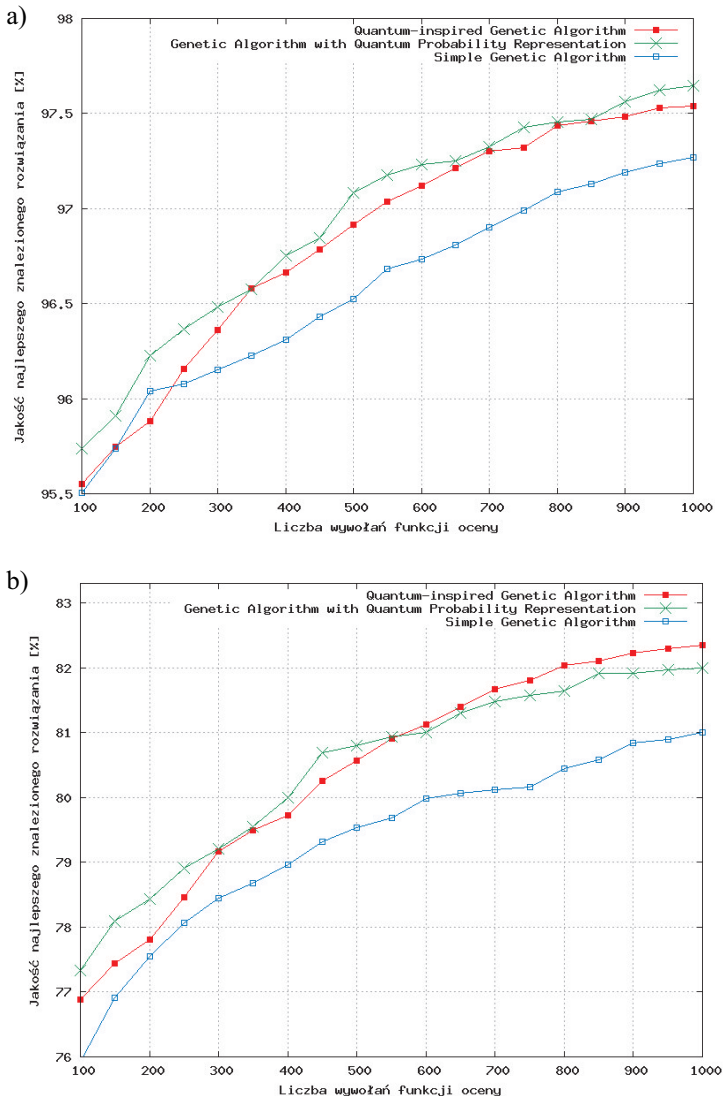
**Tabela 1**

Wyniki pracy algorytmów selekcji cech uzyskane dla tekstury sztucznej i tomograficznej przy wektorze cech o różnej długości

Obraz	Liczba cech	Algorytm	Najlepszy wynik
Tekstura sztuczna	19	GAQPR	99%
Tekstura sztuczna	19	QiGA	99%
Tekstura sztuczna	32	GAQPR	98,80%
Tekstura sztuczna	32	QiGA	98,20%
Tekstura sztuczna	48	GAQPR	98%
Tekstura sztuczna	48	QiGA	98,02%
Tekstura sztuczna	144	GAQPR	97,60%
Tekstura sztuczna	144	QiGA	97,50%
Tekstura tomograficzna	19	GAQPR	81,81%
Tekstura tomograficzna	19	QiGA	78,61%
Tekstura tomograficzna	32	GAQPR	81,40%
Tekstura tomograficzna	32	QiGA	77,47%
Tekstura tomograficzna	48	GAQPR	78,49%
Tekstura tomograficzna	48	QiGA	79,27%
Tekstura tomograficzna	144	GAQPR	84,85%
Tekstura tomograficzna	144	QiGA	84,02%

Istotną sprawą jest też rodzaj zastosowanego klasyfikatora. W przypadku klasyfikatora typu k-NN istnieje wiele jego odmian. W niniejszym opracowaniu użyto najprostszej wersji tego typu klasyfikatora, w celu zaprezentowania ogólnych możliwości takiego rozwiązania. W praktyce należy dobrać odpowiedni typ oraz parametry klasyfikatora do danego zadania.

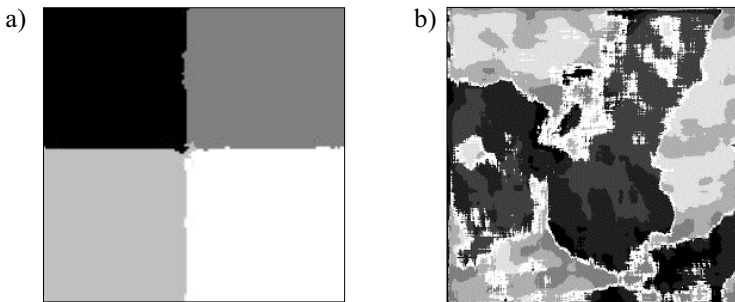
Dla dokładniejszego porównania efektywności wybranych metod na rysunku 3 zostały przedstawione wykresy, pokazujące jakość znajdowanych rozwiązań w zależności od liczby wywołań funkcji oceny. Analiza tej zależności pozwala na obiektywne porównanie badanych algorytmów, niezależnie od liczby wywołań funkcji oceny w poszczególnych generacjach algorytmów.



**Rys. 3.** Wykresy przedstawiające jakość znalezionych podzbiorów cech przez trzy porównywane algorytmy w funkcji liczby wywołań funkcji oceny, dla tekstury sztucznej (a) oraz tomograficznej (b)

We wszystkich przeprowadzonych testach algorytmy kwantowe QiGA oraz GAQPR uzyskiwały średnio lepsze rezultaty niż klasyczny algorytm genetyczny. Niewielka co do wartości różnica znalezionych rozwiązań wynika ze specyfiki zaproponowanej funkcji oceny (rys. 2). Z wykresów na rysunku 3 można ponadto odczytać szybszą zbieżność tych algorytmów.

Rysunek 4 przedstawia wyniki segmentacji tekstur, które były zaprezentowane na rysunku 1. Przedstawiony wynik uzyskano na podstawie najlepszego podzbioru cech, znalezionego przez algorytm GAQPR, odpowiednio dla tekstury sztucznej i dla tekstury tomograficznej. Jak łatwo zauważyć, segmentacja tekstury sztucznej (rys. 4a) jest całkowicie zgodna z oczekiwaniem. Poszczególne klasy są dobrze odwzorowane na posegmentowanym obrazie. Gorszą jakość segmentacji uzyskano w przypadku tekstury tomograficznej (rys. 4b). Jest to spowodowane skomplikowanym wzorcem każdej z klas, a także wysokim poziomem szumów.



**Rys. 4.** Rezultaty uzyskanej segmentacji dla tekstury sztucznej (a) i tomograficznej (b)

Następnie sprawdzono, jak przedstawione metody ewolucyjne radzą sobie z selekcją cech na tle dwóch popularnych metod wykorzystywanych dotychczas. W tym celu autorzy zaimplementowali algorytmy SBS oraz SFS [8], a następnie przetestowali ich działanie na pełnym 144-elementowym zbiorze cech i teksturze tomograficznej. Wyniki uzyskane za pomocą metod klasycznych okazały się znacznie gorsze od wszystkich porównywanych w tym artykule metod ewolucyjnych. Wartość funkcji oceny uzyskanego zbioru cech nie przekroczyła 71%. Jednakże zupełnie inna specyfika działania metod SBS i FSB uniemożliwia ich proste porównanie z metodami ewolucyjnymi. Największym problemem, a zarazem ograniczeniem algorytmów SBS i FSB jest konieczność zdefiniowania długości wektora cech, jaki ma zostać znaleziony. Analiza wyników z kilkunastu uruchomień kwantowych algorytmów genetycznych pokazała, że średnio wybierany jest wektor zawierający 65 cech i właśnie taką wartość przyjęto, testując algorytmy klasyczne. Ponadto metody klasyczne SBS i SFS działały znacznie wolniej niż prezentowane algorytmy genetyczne. Powyższe eksperymenty wykazały celowość stosowania rozwiązań heurystycznych w problemie selekcji cech.

## 6. Wnioski

W artykule został przeanalizowany problem automatycznej selekcji cech dla klasyfikatora minimalnoodległościowego  $k$ -NN w zastosowaniu do problemu segmentacji obra-

zów. Zagadnienie selekcji cech zostało sformułowane w postaci problemu optymalizacji kombinatorycznej. W artykule porównano skuteczność trzech populacyjnych metod heurystycznych w znajdowaniu optymalnego podzbioru cech. Porównane metody to prosty algorytm genetyczny oraz algorytmy genetyczne, czerpiące inspirację z systemów informatyki kwantowej: QiGA oraz GAQPR. Ocena wyników została przeprowadzona na podstawie eksperymentów z segmentacją tekstur sztucznych oraz tomograficznych.

W przeprowadzonych testach wyniki uzyskiwane przez algorytmy genetyczne, posiadające elementy inspirowane systemami informatyki kwantowej, były za każdym razem lepsze od wyników klasycznego algorytmu genetycznego. Interesującym przedmiotem dalszych badań będzie analiza wpływu parametrów przedstawionych zmodyfikowanych algorytmów genetycznych na szybkość i jakość otrzymywanych rezultatów, a także porównanie tych algorytmów dla inaczej zdefiniowanej funkcji oceny.

## Literatura

- [1] Han K.H., Kim J.H., *Genetic quantum algorithm and its application to combinatorial optimization problem*. Proceedings of the 2000 Congress on Evolutionary Computation, 2000, 1354–1360.
- [2] Han K.H., Kim J.H., *Analysis of quantum-inspired evolutionary algorithm*. Proceedings of the 2001 International Conference on Artificial Intelligence, 2001, 727–730.
- [3] Bin L., Junan Y., Zhenquan Z., *GAQPR and its application in discovering frequent structures in time series*. Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, 2003.
- [4] Nowotniak R., *Informatyka kwantowa*. XV Konferencja Sieci i Systemy Informatyczne, Łódź, 2007.
- [5] Nedjah N., Coelho L.d.S., Mourelle L.d.M., *Quantum Inspired Intelligent Systems*. Springer Verlag, 2008, 156.
- [6] Program MaZda, [http://www.eletel.p.lodz.pl/programy/cost/progr\\_mazda.html](http://www.eletel.p.lodz.pl/programy/cost/progr_mazda.html).
- [7] Strzelecki M., Materka A., *Ilościowa analiza tekstury obrazów tomograficznych rezonansu magnetycznego*. Światowy Kongres: Zastosowanie osiągnięć technologii i nauki w medycynie, 1999.
- [8] Theodoridis S., *Pattern Recognition*. 1999.
- [9] Rudnicki Z., Daca J., *Komputerowa analiza obrazów metalograficznych w badaniach warstwy wierzchniej oczek ciągadeł*. Metody i systemy komputerowe w badaniach i projektowaniu inżynierskim: IV Krajowa Konferencja, Oprogramowanie Naukowo-Techniczne, Kraków, 2003, 361–366.
- [10] Babout L. *et al.*, *Towards the texture segmentation of X-ray tomography images of lamellar microstructure in titanium based alloys*. 5th International Symposium on Process Tomography in Poland. Proceedings, 2008.
- [11] Babout L., Marrow T.J., Preuss M., *Sequential X-ray Tomography Studies of Damage Assessment in Materials Science*. 4th International Symposium on Process Tomography in Poland. Proceedings, 2006, 159–162.

- 
- [12] Rudnicki Z., Mruk M., *Detection of anisotropy of friction surface images*. VI konferencja CMS'07, Computer Methods and Systems, 2007, 271–276.
  - [13] Jóźwik A., *A recursive method for the investigation of the linear separability of two sets*. Pattern Recognition, 16(4), 1983, 429–431.
  - [14] Tschirren J., Hoffman E.A., McLennan G., Sonka M., *Intrathoracic Airway Trees: Segmentation and Airway Morphology Analysis from Low-Dose CT Scans*. IEEE Transactions on Medical Imaging, 24(12), 2005, 1529–1539.
  - [15] Nielsen M.A., Chuang I.L., *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.