

Ewa CZARNIK¹
Paweł DYMORA²
Miroslaw MAZUREK³

MECHANIZMY WYSZUKIWANIA OBRAZEM W ORACLE 11g

W bazach danych zawierających dane graficzne (np. zdjęcia), niejednokrotnie zachodzi potrzeba szybkiego odnalezienia podobnego obrazu. W bazach danych np. firm magazynowych, ważną kwestią może być eliminacja duplikującego się asortymentu posiadającego różne opisy, dokumentację, a faktycznie posiadającego te same właściwości fizyczne, np. poprzez porównanie wyglądu poszczególnych produktów na podstawie posiadanego zdjęcia. Naprzeciw takim potrzebom wychodzi Oracle ze standardem SQL/MM, który udostępnia metody umożliwiające przeszukiwanie baz danych za pomocą właściwości wizualnych tzw. wyszukiwanie obrazem. Artykuł prezentuje możliwości technologii Oracle 11g obsługującej typy składowania danych multimedialnych (w tym danych graficznych) oraz przykład stworzonej aplikacji internetowej umożliwiającej implementację tych mechanizmów i wyszukiwanie obrazem. Aplikacja została wykorzystana do przeprowadzenia badań wydajnościowych różnych metod wyszukiwania obrazem.

Słowa kluczowe: Oracle 11g, SCORE, wyszukiwanie obrazem.

1. Multimedialne bazy danych

Coraz częściej zachodzi potrzeba zapisywania dużej ilości danych w bazach danych nie tylko o charakterze tekstowym, ale także multimedialnym. Naprzeciw temu wychodzą multimedialne rozszerzenia baz danych, które zostały stworzone z myślą o przechowywaniu obrazów, dźwięków, wideo oraz dużych

¹ Autor do korespondencji: Ewa Czarnik, Politechnika Rzeszowska, adres e-mail: ewa1927@gmail.com

² Paweł Dymora, Politechnika Rzeszowska, Zakład Systemów Złożonych, pawel.dymora@prz.edu.pl

³ Miroslaw Mazurek, Politechnika Rzeszowska, Zakład Systemów Złożonych, mirosław.mazurek@prz.edu.pl

dokumentów tekstowych. W artykule skupiono się na mechanizmach związanych z danymi graficznymi i obrazami.

Multimedialne bazy danych pozwalają użytkownikowi na przechowywanie, zarządzanie oraz wykonywanie operacji na plikach multimedialnych. RDBMS Oracle wspiera m.in. standard Oracle Multimedia, który umożliwia przeszukiwanie bazy danych oraz analizę danych w niej zawartych nie tylko pod kątem podstawowych typów danych (znakowe, numeryczne, data/czas), ale przede wszystkim danych multimedialnych. Jej główną zaletą jest możliwość wyszukania danych podobnych do tych, które już posiadamy. Istnieją następujące sposoby przeszukiwania bazy danych w odniesieniu do typów multimedialnych [1-3]:

- za pomocą liczbowych oraz tekstowych parametrów (nazwa, ID, producent),
- podzapytanie, w którym umieścimy podobnie brzmiący fragment ścieżki dźwiękowej, bądź podobny obraz.

1.1. Oracle Multimedia

Oracle Multimedia to funkcjonalność serwera bazodanowego Oracle (dostępna od wersji 10.2), oparta o mechanizm obiektowo-relacyjny, dostarczająca nowe typy danych, dzięki którym możemy przechowywać i zarządzać danymi multimedialnymi zawartymi w bazie danych. Typy danych Oracle Multimedia (typy obiektowe zdefiniowane w schemacie ORDSYS, a składowane w bazie jako BLOB) to [1, 4, 5]:

- ORDAudio – składowanie i przetwarzanie obiektów audio;
- ORDVideo – składowanie i przetwarzanie obiektów wideo;
- ORDImage – składowanie i przetwarzanie obrazów;
- ORDDoc – składowanie heterogenicznych obiektów multimedialnych.

Typ ORDAudio dotyczy możliwości przechowywania plików dźwiękowych m.in. w następujących formatach: *.3gp*, *.aff*, *.mpg*, *.wav* i inne. Każdy z tych formatów posiada własne informacje rozpoznawalne przez bazę danych, takie jak: *ID formatu*, *format*, *rozszerzenie* czy *typ MIME*. Każdy obiekt zapisany w tym formacie posiada atrybuty dotyczące typu danych oraz samego pliku dźwiękowego. Do pierwszej kategorii zaliczamy m. in. *opis*, *źródło danych* oraz *typ MIME*, a do drugiej zastosowany *typ kompresji*, *całkowitą długość pliku* czy *typ dekodowania* [2, 4].

Kolejny typ to ORDDoc, który dotyczy heterogenicznych plików zawierających dużą ilość danych (np. grafika, audio, wideo). W odróżnieniu od ORDAudio nie zawiera danych dotyczących pliku, a jedynie te opisujące typ danych [2, 5].

Typ `ORDImage` służy do przechowywania plików graficznych w popularnych formatach takich jak: `.bmp`, `.gif`, `.jpg`, `.png`. Również w tym przypadku zrezygnowano z atrybutów dotyczących pliku, a atrybuty opisujące typ zostały rozszerzone o m. in. *wysokość* oraz *szerokość obrazu* w pikselach, format obrazu [2, 5].

Ostatni z wprowadzonych typów `ORDVideo`, dotyczy plików wideo. Atrybuty opisujące ten typ powstały poprzez połączenie i zmodyfikowanie atrybutów dotyczących typów `ORDImage` oraz `ORDAudio`. Znajdziemy w nich pola takie jak: *szerokość* i *wysokość w pikselach*, *typ kompresji*, ale również *szybkość transmisji*, *ilość klatek* oraz *liczbę kolorów użytych w wideo* [2].

Typy danych `ORDAudio`, `ORDVideo`, `ORDImage` i `ORDDoc` dostępne były również w starszych wersjach serwera bazy danych Oracle. W Oracle 10g, w ramach rozszerzenia Oracle Multimedia, wprowadzono nowe typy danych `SI_StillImage` (*Still Image*), umożliwiające zgodne ze standardem SQL/MM operacje na obrazach w bazie danych (alternatywę dla `ORDImage`) [2, 3, 5].

1.2. Standard SQL/MM

Począwszy od wersji Oracle 10g został wprowadzony standard nazywany SQL/MM (ang. *SQL Multimedia and Applications Package*). Został on zaprojektowany z myślą o multimedialnych bazach danych, ale również o specjalistycznych zastosowaniach systemów bazodanowych. Pierwsza edycja standardu pochodzi z roku 2001 (ISO/IEC 13249-5:2001) [3]. Standard składa się z kilku części, które w porównaniu do tradycyjnego SQL, są ze sobą dość luźno powiązane. Pierwsza część wchodząca w skład SQL/MM nosi nazwę *Framework*. Służy ona za podstawę definicji standardu dla pozostałych części. Zawiera informacje dotyczące zakresu standardu, definicje i koncepcje wspólne dla wszystkich części standardu oraz tego, w jaki sposób elementy wykorzystują mechanizmy SQL. Kolejne fragmenty to: *Full-Text*, *Spatial* oraz *Still Image*, dotyczą kolejno tekstowych i przestrzennych baz danych oraz baz, które zawierają obrazy [2, 3, 5]. Warto zwrócić uwagę na fakt, iż w implementacji została pominięta część nr 4, która miała dotyczyć działań matematycznych, lecz prace nad nią zostały po pewnym czasie zawieszono. Ostatnie dwa elementy składowe, czyli część nr 6 odpowiedzialna za *Data Mining* i część nr 7, czyli moduł *History*, dotyczą specjalistycznych zastosowań bazy danych. Odnoszą się one do eksploracji danych i przetwarzania danych historycznych. W implementacji zabrakło miejsca na typy danych, które dotyczą plików audio i wideo, mimo iż nazwa bezpośrednio nawiązuje do plików multimedialnych [4]. Warto podkreślić, że od wersji Oracle 10g mamy implementację specyfikacji standardu SQL/MM tylko w zakresie obrazów - SQL/MM *Still Image* [2, 3, 5].

2. Metadane

Każdy obiekt multimedialny posiada pewne dane o swoich właściwościach fizycznych i logicznych. Metadane można podzielić na dwie kategorie: w podejściu bibliotekoznawczym i podejściu informatycznym. W tym ostatnim metadane wykorzystuje się w celu zarządzania określonymi danymi, a podstawowym ich zadaniem jest dostarczenie uporządkowanej i logicznej dokumentacji, która opisuje sposób wykorzystania i powstania danych [1, 5].

Metadane to dodatkowe informacje o danych takie jak *wielkość danych*, *typ kompresji* czy *format* [1, 4, 5], służą one do rozróżnienia poszczególnych aspektów opisywanych danych. Dzięki nim możemy pozyskać takie informacje jak np. *data pozyskania danych*, *informacje dotyczące autora* oraz *praw autorskich*, w jakich *formatach* są one dostępne i wiele innych. Główną zaletą takich danych jest łatwość w odczycie oraz analizie, ze względu na fakt, iż są one zapisane przy pomocy składni języka XML. Prosty przykładem metadanych może być księgozbiór w bibliotece. Katalog biblioteczny zawiera informacje o autorach, datach publikacji lub wydaniach itp. Kolejnym walorem takich danych jest fakt, że mogą one opisywać nie tylko dokumenty tekstowe, lecz także graficzne (tj. obrazy, grafika), dźwiękowe oraz wideo.

2.1. Metadane w Oracle

Wszystkie metadane zawarte są w plikach multimedialnych. System bazodanowy Oracle od wersji 10g pozwala nam na generowanie oraz bezpośredni dostęp do metadanych plików multimedialnych. Funkcja metadanych zwiększa możliwości obiektów typu Oracle Multimedia dodając możliwość m. in. zapisywania oraz odczytywania rozszerzonych informacji np. z obrazów.

Dzięki temu, iż zostało udostępnionych kilka rodzajów metadanych, możemy wykorzystywać je w różnych celach. Pierwszym z typów są *metadane techniczne*. Opisują one parametry obrazu w sensie technicznym, tzn. mogą one opisywać *wysokość* i *szerokość* obrazu, *rodzaj kompresji* czy *format* w jakim został zapisany obiekt graficzny. Drugi z typów (*metadane wyszukiwania*) opisuje takie właściwości jak *datę* i *czas* utworzenia obrazu, jego *autora*. Trzecim, i za razem ostatnim typem, są *metadane osadzone*. Charakteryzują się one tym, że są one zapisane bezpośrednio w formacie pliku obrazu. Takie metadane są reprezentowane przez plik XML. Może on być przechowywany w bazie danych, indeksowany, przeszukiwany, aktualizowany, a także udostępniany aplikacjom wykorzystującym standardowe mechanizmy Oracle Database [3, 5].

```

1 <xmpMetadata xmlns="http://xmlns.oracle.com/ord/meta/xmp" xsi:schemaLocation=
2 "http://xmlns.oracle.com/ord/meta/xmp http://xmlns.oracle.com/ord/meta/xmp" xmlns:xsi=
  "http://www.w3.org/2001/XMLSchema-instance">
3 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:ix=
  "http://ns.adobe.com/ix/1.0/">
4 <rdf:Description about="uuid:bd81a53c-1c9d-11d9-aaba-b98a815924f6" xmlns:pdf=
  "http://ns.adobe.com/pdf/1.3/">
5 <!-- pdf:Subject is aliased -->
6 </rdf:Description>
7 <rdf:Description about="uuid:bd81a53c-1c9d-11d9-aaba-b98a815924f6" xmlns:photoshop=
  "http://ns.adobe.com/photoshop/1.0/">
8 <photoshop:Headline>Monitor</photoshop:Headline>
9 <photoshop:Credit>Oracle Corporation</photoshop:Credit>
10 <photoshop:Source>Internal Digital Camera</photoshop:Source>
11 <photoshop:City>Nashua</photoshop:City>
12 <photoshop:State>NH</photoshop:State>
13 <photoshop:Country>USA</photoshop:Country>
14 <photoshop:DateCreated>2004-10-12</photoshop:DateCreated>
15 <!-- photoshop:Caption is aliased -->
16 </rdf:Description>
17 <rdf:Description about="uuid:bd81a53c-1c9d-11d9-aaba-b98a815924f6" xmlns:xap=
  "http://ns.adobe.com/xap/1.0/">
18 <!-- xap:Description is aliased -->
19 </rdf:Description>
20 <rdf:Description about="uuid:bd81a53c-1c9d-11d9-aaba-b98a815924f6" xmlns:xapMM=
  "http://ns.adobe.com/xap/1.0/mm/">
21 <xapMM:DocumentID>adobe:docid:photoshop:186fd660-1c9c-11d9-aaba-b98a815924f6
  </xapMM:DocumentID>
22 </rdf:Description>
23 <rdf:Description about="uuid:bd81a53c-1c9d-11d9-aaba-b98a815924f6" xmlns:dc=
  "http://purl.org/dc/elements/1.1/">
24 <dc:description>
25 <rdf:Alt>
26 <rdf:li xml:lang="x-default"/>
27 </rdf:Alt>
28 </dc:description>
29 </rdf:Description>
30 </rdf:RDF>
31 </xmpMetadata>

```

Rys. 1. Przykładowe metadane w standardzie XMP (ang. Extensible Metadata Platform)

Fig. 1. Sample metadata in XMP standard

```

1 <ordImageAttributes xmlns="http://xmlns.oracle.com/ord/meta/ordimage" xmlns:xsi=
  "http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation=
  "http://xmlns.oracle.com/ord/meta/ordimage http://xmlns.oracle.com/ord/meta/ordimage">
2 <height>225</height>
3 <width>300</width>
4 <contentLength>21491</contentLength>
5 <fileFormat>JFIF</fileFormat>
6 <contentFormat>24BITRGB</contentFormat>
7 <compressionFormat>JPEG</compressionFormat>
8 <mimeType>image/jpeg</mimeType>
9 </ordImageAttributes>

```

Rys. 2. Przykładowe metadane typu ORDImage

Fig. 2. Sample metadata for ORDImage type

Rys. 1 przedstawia przykładowe metadane typu EXIF. Jest to standard służący przechowywaniu plików graficznych stworzonych przez cyfrowe aparaty fotograficzne. Został opracowany przez japońskie stowarzyszenie JEIDA (ang. *Japan Electronic Industry Development Association*) jako standard przechowywania zdjęć zrobionych aparatem cyfrowym i metadanych o danym obrazie. Oracle Multimedia wspiera przechowywanie metadanych tego typu i ich ekstrakcję z grafiki o formatach JPEG i TIFF.

Metadane zapisane formacie EXIF czy TIFF, są opisywane zgodnie ze standardem XMP, co ułatwia ich przeglądanie i zarządzanie nimi.

Rys. 2 pokazuje przykładowe metadane typu ORDImage. Ten typ metadanych został wprowadzony przez firmę Oracle jako część funkcjonalności Oracle Multimedia. Przechowuje on najbardziej podstawowe informacje o obrazie, takie jak *wysokość* i *szerokość*, *format* w jakim plik został zapisany i itp.

3. Wyszukiwanie obrazów w oparciu o zawartość

Przechowywanie danych multimedialnych niesie ze sobą wiele wyzwań. Dotychczasowe modele danych, jak również dostępne języki zapytań w znikomym stopniu odnoszą się do złożonej charakterystyki danych multimedialnych. Ważnym zagadnieniem nad którym prowadzone są prace, stała się możliwość wyszukiwania obrazów poprzez porównywanie wzorcowego obrazu z tymi zamieszczonymi w bazie danych. Możliwe jest wyszukiwanie podobieństw obrazów za pomocą metadanych, niemniej ta technika może być ograniczona jedynie do danych opisowych obrazów. Istotą zagadnienia jest wczytanie wzorca obrazu i odszukanie innych obrazów będących niejako kopią tego wzorca lub jego najwierniejszym odwzorowaniem. Wyszukiwanie obrazem polega na wpisaniu polecenia, które ma za zadanie stworzenie modelu obrazu w oparciu o zawartość graficzną np. na podstawie średniego koloru bądź histogramu rozkładu kolorów na obrazie. Następnie do testowania podobieństwa obrazów można zastosować metodę SI_SCORE dostępną dla typu danych multimedialnych SI_STILLImage standardu SQL/MM [1, 4, 5].

CBIR (ang. *Content-based Image Retrieval*) proponuje specjalny model reprezentacji zawartości obrazów. Każdy z obrazów w bazie danych jest opisany przez zmodyfikowany diagram encji-związków. Encje nie oznaczają tu jednak typów, ale konkretne obiekty. Podobnie symbol związku dotyczy jednego konkretnego powiązania, a nie zbioru powiązań. SCORE najlepiej stosować, gdy szukany obraz zajmuje cały lub większość obszaru dostępnych obrazów [2, 5]. Dzięki temu wyniki wyszukiwania są bardziej precyzyjne. Jednakże w przypadku gdy na obrazie zawarty jest wiele różnych kształtów system efektywniej poradzi sobie z wyszukaniem jeżeli będą one mało skomplikowane oraz w kontrastujących kolorach. Niestety nie dorównuje on człowiekowi w zakresie przeszukiwania, ponieważ ludzkie oko potrafi dostrzec podobieństwo na obrazach, gdzie szukany obiekt jest znacznie przeskalowany, bądź lekko przesłonięty przez inny kształt. Dlatego też należy wziąć pod uwagę to, że przeszukiwanie takie jest miarodajne tylko we wstępnym etapie wyszukiwania, kiedy zachodzi potrzeba odizolowania większej ilości niepodobnych obrazów [1, 4].

Metoda `SI_SCORE` zwraca nieujemną liczbę zmiennoprzecinkową, a poszczególne wartości określają stopień dopasowania wzorca. Im mniejsza wartość tym lepsze dopasowanie do wzorca, a wartość 0 to najlepsze dopasowanie do wzorca. Użycie funkcji `SI_SCORE` jest wymuszone, gdyż typ `SI_STILLImage` nie posiada metod do bezpośredniego porównywania dwóch obiektów graficznych. Niemniej standard SQL/MM dostarcza nam kilku metod dzięki którym w dość prosty sposób możemy określać właściwości wizualne obrazów, które następnie mogą być wykorzystane do porównania z obrazem źródłowym za pomocą metody `SI_SCORE` [2, 5]. Są to:

- `SI_FindClrHstgr`,
- `SI_FindPstnlClr`,
- `SI_FindAvgClr`,
- `SI_FindTexture`.

Pierwsza z metod dotyczy histogramu koloru. Algorytm w zastosowanej metodzie polega na tym, iż w pierwszym kroku obraz jest dzielony na obszary o stałej wielkości. Każdy z takich obszarów obejmuje zbiór kolorów i jest reprezentowany przez jeden z kolorów. Następnie wyznacza się dla każdego z nich częstotliwości ich występowania poprzez iterację po pikselach. Po wykonaniu tego kroku należy znormalizować wartość częstotliwości, aby zawierała się ona w przedziale od 0 do 100. Histogram jest zapamiętywany jako sekwencja koloru, częstotliwości, lecz fizycznie jest zapisywany w postaci tablic, gdzie pierwsza z nich przedstawia kolor, a druga częstotliwości występowania danego koloru.

Algorytm dotyczący pozycji kolorów (`SI_FindPstnlClr`) to algorytm, który służy do wyznaczania lokalizacji kolorów. Również i w tym przypadku obraz jest dzielony na pewne obszary. Po wyznaczeniu obszarów wyznaczany jest dla nich kolor, który występuje najczęściej. Jest on wyznaczany na podstawie histogramu kolorów. Wynik, czyli lokalizacja kolorów to tablica obiektów typu `SI_Color`.

W metodzie `SI_FindAvgClr` komponenty kolorów składowych: czerwonego, zielonego i niebieskiego, z każdej ze stworzonych próbek z obrazu są sumowane, a następnie dzielone przez ilość próbek. Wynik jest reprezentowany przez właściwość typu `SI_Color`.

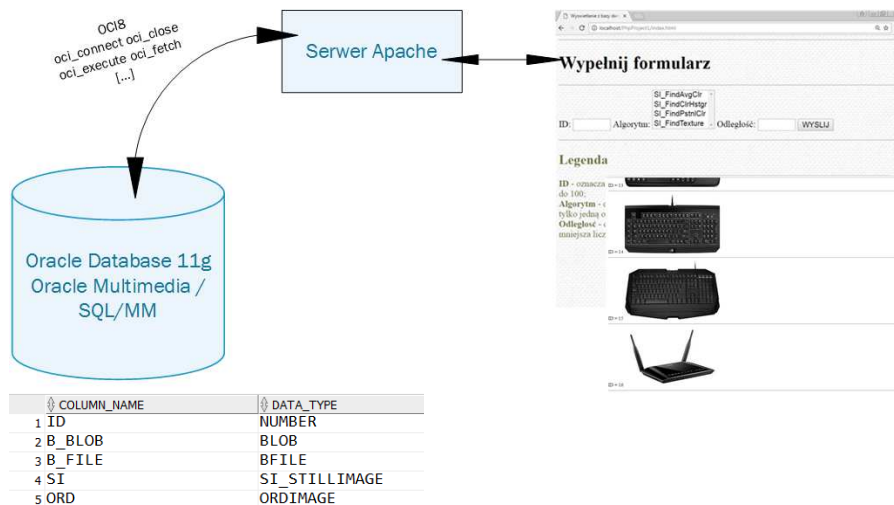
Ostatni z przytoczonych algorytmów `SI_FindTexture` jest jedynym, który nie został ściśle opisany przez standard. Zasada jego działania jest zależna od implementacji. Opiera się on o typ danych `SI_Texture`, który opisuje teksturę obrazu charakteryzującą się fakturą materiału (chropowata, gładka), jasność, kontrast i inne.

4. Badanie wydajności

W niniejszym rozdziale przedstawiono rezultaty testów dotyczące efektywności mechanizmów wyszukiwania obrazem w bazie danych Oracle implementującej standard Oracle Multimedia. Testy zrealizowano w oparciu o przygotowany system wyszukiwania w bazie danych, który składał się z RDBMS Oracle Database 11g, serwera Apache oraz aplikacji internetowej. Schemat implementacji zaproponowanego rozwiązania wykorzystującego mechanizmy wyszukiwania obrazem poddanemu badaniu przedstawiono na Rys. 3.

4.1. Środowisko testowe

Testowa baza danych zawierała 100 wierszy, które m.in. posiadały pole przechowujące obraz oraz jego nazwę. Dla każdego z nich wykonano 4 podzapytania w zależności od obranej dokładności współczynnika podobieństwa: 1, 5, 10, 50; im większa odległość podobieństwa tym mniej podobny obraz. Badanie zostało przeprowadzone dla każdej z czterech metod (SI_FindClrHstgr, SI_FindPstnlClr, SI_FindAvgClr, SI_FindTexture), co oznacza, że każdy obraz został przeanalizowany 16 razy. Podzapytania były wykonywane w środowisku SQL Developer dzięki czemu możliwe było zapisanie wyników wyszukiwania. Skupiono się w szczególności na ilości zwracanych wierszy oraz średnim czasie wykonywania zapytania.



Rys. 3. Model implementacji środowiska testowego

Fig. 3. Model of the test environment implementation

Podstawowym problemem wydajnościowym w wyszukiwaniu obrazem jest rozmiar obrazu jakim wyszukujemy. Algorytm aplikacji testowej został zaprojektowany tak, aby wszystkie obrazy były skalowane do jednego rozmiaru. W wyniku tego może się okazać, że obrazy bardzo podobne po takiej operacji okażą się niewystarczająco podobne i nie zostaną zwrócone jako rezultat zapytania. Należy również pamiętać o tym, aby zachować jeden schemat kolorowania obrazów. W przypadku analizowania dwóch obrazów, jednego czarno-białego i drugiego kolorowego, z pewnością okaże się, iż są to dwa różne obrazy. Kolejnym kluczowym problemem jest sposób oświetlenia przedmiotu na obrazie. W sytuacji, gdy zastosujemy różne oświetlenie dla jednego obiektu może się okazać, iż na jednym zdjęciu obiekt ten będzie bardzo dobrze oświetlony i dokładnie widoczny, a na drugim zaciemniony przez co system nie będzie w stanie poprawnie go przeanalizować.

Przygotowana aplikacja testowa do badania mechanizmów wyszukiwania obrazem, składa się z dwóch głównych plików: `index.html` oraz `index.php`. W pierwszym z nich zawarty jest formularz, który umożliwia wprowadzenie trzech danych: *id obrazka*, *wybrana metoda wyszukiwania* oraz *odległość podobieństwa* do wyszukiwania obrazem w bazie danych. Plik HTML zawierał w sobie formularz z polami do wyboru.

Wypełnij formularz

ID: Algorytm: Odległość: WYSLIJ

Legenda

ID - oznacza ID obrazka, dla którego wyszukujemy obraz podobny; poprawne wartości to od 1 do 100;

Algorytm - oznacza wybrany algorytm wyszukiwania obrazka podobnego, można wybrać tylko jedną opcję;

Odległość - oznacza odległość podobieństwa obrazów; poprawne wartości to od 1 do 100; mniejsza liczba to większa dokładność;

Rys. 4. Formularz główny aplikacji

Fig. 4. Main application form

Jak można zaobserwować na Rys. 4 użytkownik ma możliwość wyboru *ID obrazka*, *algorytm wyszukiwania* oraz *odległość podobieństwa*. Po wybraniu odpowiednich parametrów oraz naciśnięciu przycisku WYSLIJ formularz przesyła dane poprzez zapytania SQL do bazy danych. Po realizacji zapytania

zwracane są na stronie WWW rezultaty spełniające warunek podobieństwa. Przykładowe rezultaty testów aplikacji wyszukiwania obrazem w multimedialnej bazie danych zostały przedstawione na Rys. 5 oraz Rys. 6.

Wybrane opcje

ID: 5
 Algorytm: SI_FindAvgClr
 Odległość: 1

```
select p1.id,f.name from photos p1, files f where f.id=p1.id and SI_FindAvgClr((select p2.si from photos p2 where id=5)).SI_score(p1.si)<1
```

Wyniki wyszukiwania



ID = 5

Rys. 5. Przykładowe wyniki wyszukiwania obrazem

Fig. 5. Sample results of image reverse search



ID = 4



ID = 16



ID = 23



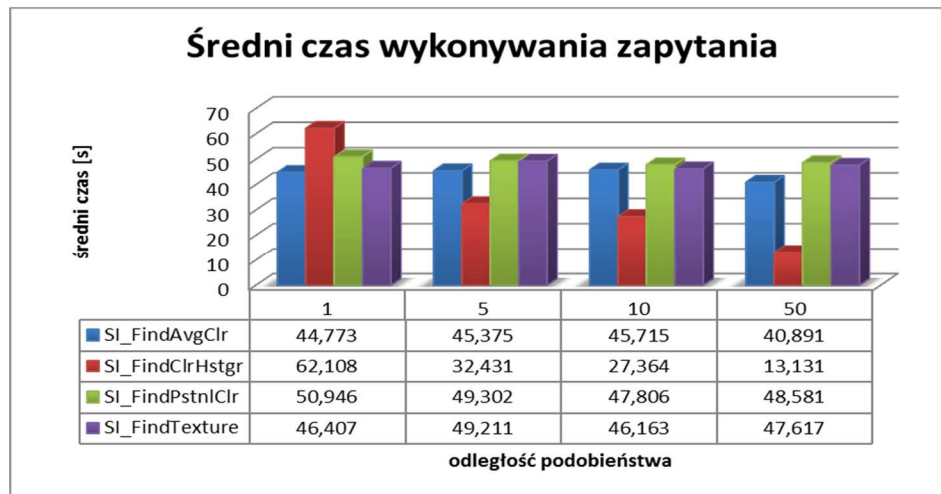
Rys. 6. Przykładowe wyniki wyszukiwania obrazem

Fig. 6. Sample results of image reverse search

4.2. Rezultaty testów

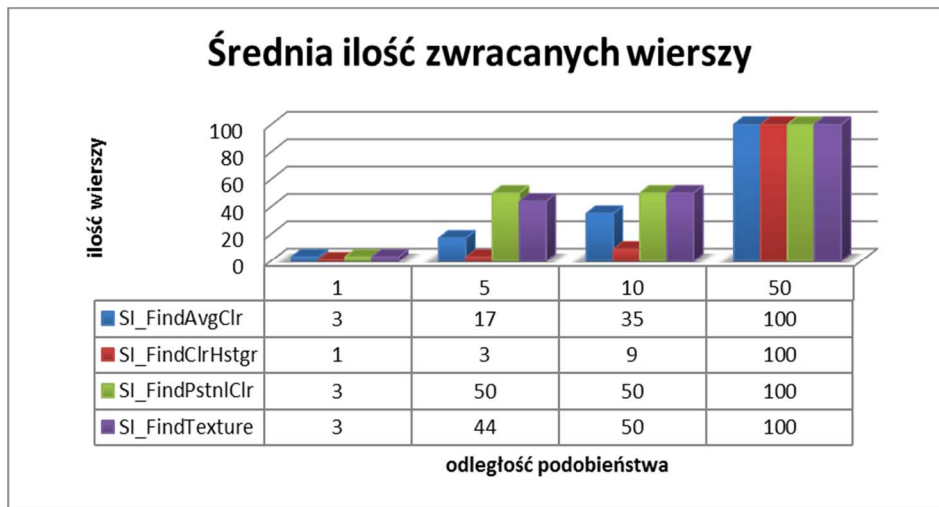
Po przeprowadzeniu szeregu testów można było zauważyć, że najbardziej efektywną metodą wykorzystywaną do wyszukiwania obrazem jest metoda opierająca się na histogramie kolorów. Próby pokazały, iż w niemal każdym przypadku jest to najszybszy algorytm, a przy tym zwraca najmniejszą ilość wierszy, czyli bardziej dokładny wynik odwzorowania. Analiza wyników pokazała, że średni czas wszystkich wykonywanych zapytań to 33,758 s, a średnia ilość wierszy to 29. Na drugim miejscu uplasowała się metoda SI_FindAvgClr, opierająca się na średnim kolorze. Średni czas wykonywania poleceń to 44,186 s a średnia ilość zwracanych wierszy to 39. Kolejnym algorytmem jest SI_FindTexture, który bazuje na teksturze obrazu. Średni czas i średnia ilość wierszy to odpowiednio 47,350 s i 49. Najmniej wydajnym algorytmem okazał się ten, który bazuje na pozycji danego koloru, czyli SI_FindPstnlClr. Jego średnie wyniki w testach to 49,159 s i 51 wierszy. Różnica pomiędzy tym algorytmem, a algorytmem najefektywniejszym wynosi 15,4 s i 23 wiersze. W przypadku niewielkiej ilości rekordów w testowej bazie danych ta różnica może się wydawać niewielka, lecz w przypadku gdy baza danych będzie magazynować setki tysięcy obrazów, różnica taka może się znacznie powiększyć i skutecznie utrudnić pracę użytkownika.

Rys. 7 oraz Rys. 8 przedstawiają wyniki wyszukiwania dla każdej metody z podziałem na odległości podobieństwa, natomiast dwa kolejne (Rys. 9 oraz Rys. 10) bez podziału na odległość. Przedstawiono uśrednione wyniki ze wszystkich zapytań.



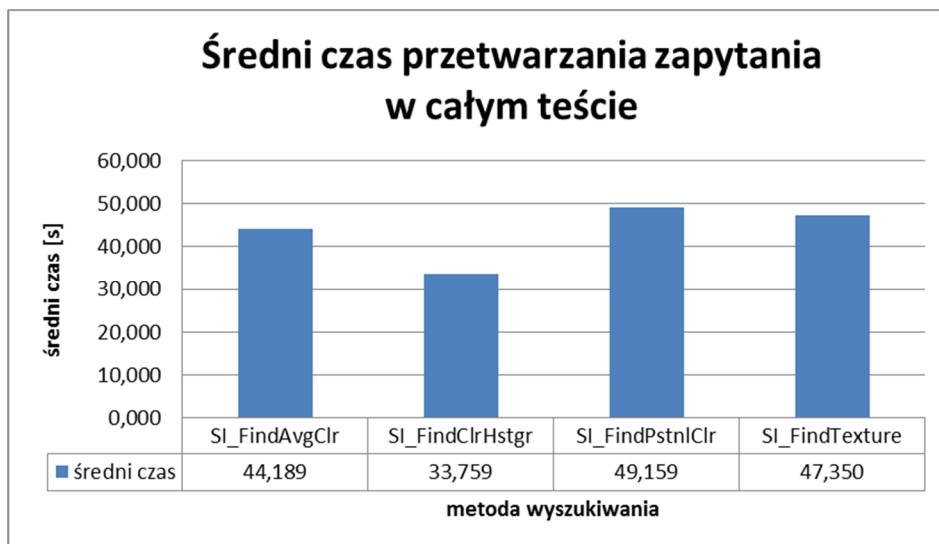
Rys. 7. Wykres przedstawiający średni czas wykonywania zapytania

Fig. 7. Average query execution time



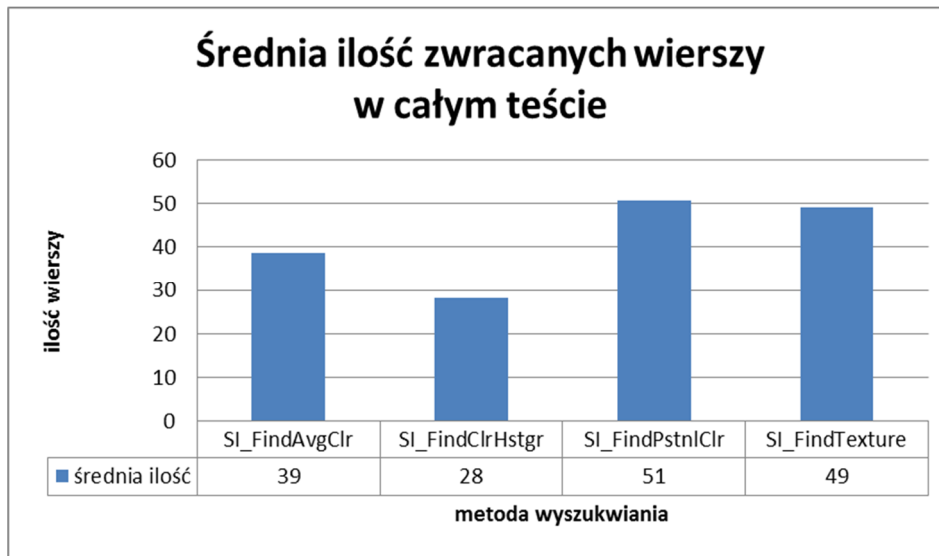
Rys. 8. Wykres przedstawiający średnią ilość zwracanych wierszy

Fig. 8. Average number of returned rows



Rys. 9. Wykres przedstawiający średni czas wykonywania zapytania w całym teście

Fig. 9. Average query execution time in the whole test



Rys. 10. Wykres przedstawiający średnią ilość zwracanych wierszy w całym teście

Fig. 10. Average number of returned rows in the whole test

5. Podsumowanie

Obecnie bazy danych coraz częściej, tuż obok tekstu, liczb, czy obiektów data/czas, zawierają obiekty multimedialne, takie jak obrazy, audio oraz wideo. Niejednokrotnie zachodzi potrzeba szybkiego odnalezienia podobnego rekordu danych tylko na podstawie wprowadzanego do systemu zdjęcia, bez podawania innych opisujących go atrybutów. Naprzeciw takim potrzebom wychodzi RDBMS Oracle z implementacją standardu SQL/MM, który udostępnia metody umożliwiające przeszukiwanie bazy danych za pomocą wskaźników odnoszących się do właściwości wizualnych obiektów multimedialnych. Aby efektywniej wyszukiwać obrazy należy dobrać odpowiednią wartość wag tych parametrów. Niestety odpowiedni dobór takiej wartości jest bardzo trudny, ponieważ należy to do subiektywnej oceny użytkownika. Należy pamiętać szczególnie o tym, że komputer nie dorównuje ludzkiemu oku w kwestii porównywania obrazów, jednakże może mu zdecydowanie pomóc.

W testowanej bazie danych odizolowanie nawet 50% obrazów niespełniających kryterium podobieństwa znacząco skraca czas ewentualnego „ręcznego” przeszukiwania bazy danych. Kluczowym elementem całego mechanizmu jest oczywiście metoda `SI_Score`, która odpowiada za próg podobieństwa. Wyższe wartości reprezentują mniejszą dokładność wyszukiwania, a mniejsze – lepszą dokładność. Podczas doboru tego parametru należy zadać sobie pytanie co jest krytyczną wartością systemu. Jeżeli

użytkownikom zależy na szybszym wykonaniu zapytań, należy zwiększyć wartość parametru odpowiedzialnego za podobieństwo. Jednakże w przypadku, gdy chcemy otrzymać obrazy bardzo podobne do szukanego należy zmniejszyć ten parametr.

Rezultaty przeprowadzonych badań w znaczący sposób mogą podnieść efektywność rozwiązań wykorzystujących mechanizmy wyszukiwania obrazem poprzez lepsze dobranie odpowiednich mechanizmów przez projektantów aplikacji i administratorów baz danych.

Literatura

- [1] Górnicki M.: *Wyszukiwanie obrazów na podstawie zawartości*, 2014.
- [2] *Oracle Multimedia Reference 11g Release 2*, <https://docs.oracle.com/> [dostęp 10-04-2017].
- [3] *ISO/IEC 13249-5:2003, Information Technology – Database Languages – SQL Multimedia and Application Packages – Part 5: Still Image*. ISO, 2003.
- [4] Bryła B., Loney K.: *Oracle Database 11g. Podręcznik Administratora*, Helion, 2013.
- [5] *Oracle Multimedia User's Guide 11g Release 2*, <https://docs.oracle.com/> [dostęp 10-04-2017].

REVERSE IMAGE-SEARCHING MECHANISMS IN ORACLE 11g

Summary

The main aim of this paper is to present mechanisms and types of multimedia data storage in Oracle 11g Database System along with a web application that allows reverse image search mechanism implementation in order to test its efficiency. First part shows Oracle Multimedia and SQL/MM standard, with supported data types, metadata and methods. The second part is focused on presentation the original database application with appropriate tables and procedures implementing reverse image search. The application with appropriate database is used to determinate on the base of series tests the most efficient image search methods.

Keywords: Oracle Multimedia, SQL/MM, images, reverse image searching, SCORE

DOI: 10.7862/re.2017.13

Tekst złożono w redakcji: wrzesień 2017

Przyjęto do druku: październik 2017