STUDIA INFORMATICA 2013
Volume 34 Number 2A (111)

Marek BANACH, Krzysztof CZAJKOWSKI, Tomasz GĄCIARZ
Instytut Teleinformatyki, Politechnika Krakowska

WYSZUKIWANIE I ANALIZA FOTOGRAFII OBIEKTÓW ZABYTKOWYCH W SCIECI INTERNET

Streszczenie. Artykuł prezentuje autorską propozycję systemu wyszukiwania w sieci Internet treści i fotografii dotyczących obiektów zabytkowych, mogącego stanowić wsparcie dla prac konserwatorskich. Zaprezentowano możliwy zestaw cech, których uwzględnienie pozwala selekcjonować fotografie pod kątem ich przydatności w omawianym zagadnieniu.

Słowa kluczowe: wyszukiwarka internetowa, analiza obrazu, zabytek

SEARCHING AND ANALYSING PHOTOS OF HISTORICAL MONUMENTS IN THE INTERNET NETWORK

Summary. This paper presents author's proposal of solutions for searching contents and photos about historical monuments in the Internet. The aim of this system is to support conservation works. The possible set of features, which can be used in photos selection from the point of view of their usefulness, has been presented.

Keywords: search engine, image analysis, historical monument

1. Wstęp

Ochrona zabytków jest w dzisiejszych czasach jednym z kluczowych kierunków działań mających na celu zachowanie dziedzictwa kulturowego poszczególnych krajów. Z uwagi na złożoność problemu osoby i instytucje zaangażowane w takie działania stają przed licznymi problemami. Różnorodność obiektów uznawanych za zabytki oraz bardzo duży przedział czasowy powstawania poszczególnych struktur powodują, że prowadzenie prac konserwatorskich jest dużym wyzwaniem. Działania obejmujące swym zakresem przywracanie obiektom ich dawnego wyglądu wymagają często wykonywania rekonstrukcji elementów obecnie już

nieistniejących. Czasami przy pracach zabezpieczających wymagane są specjalne techniki oraz materiały. Bardzo użyteczne mogłoby być oparcie się na pracach renowacyjnych wykonywanych wcześniej na innych obiektach tego typu, o podobnej konstrukcji, pochodzących z tej samej epoki lub wykonanych z takich samych materiałów. Jeżeli obiekty o zbliżonych parametrach nie znajdują się w pobliżu, istnieje spora szansa na znalezienie właściwych budowli w innych regionach kraju lub nawet w innych krajach. Jednakże w celu dokonania takiego przeszukiwania niezbędna jest ogromna wiedza.

Liczba zabytków nieruchomych w samej tylko Polsce przekracza obecnie 65 000 i nieustannie rośnie [13, 14]. Najwłaściwszym rozwiązaniem problemu, jakim jest wyszukiwanie informacji w takich zbiorach, wydaje się skorzystanie z bazy danych o zabytkach. Pojawiają się jednak liczne trudności. Po pierwsze wiele krajów (w tym Polska) nie posiada kompletnych baz danych obiektów zabytkowych. Po drugie istniejące bazy tego typu skupiają się raczej na danych tekstowych (wymiarach, datach, opisach, właścicielach itd.), a w mniejszym stopniu kładą nacisk na treści multimedialne. Trzecim zagadnieniem jest konieczność wielopoziomowego opisu ze względu na często bardzo złożoną strukturę obiektów zabytkowych. Kolejny problem to różne konstrukcje takich systemów, struktur ich baz danych, interfejsów. Znacząco utrudnia to integracje i ich sprawne przeszukiwanie. Jeszcze inny problem to kwestia zunifikowanego sposobu formułowania zapytań i uzyskiwanych rezultatów.

Uzasadniona wydaje się próba wykorzystania sieci Internet, stanowiącej ogólnodostępne źródło informacji. Na stronach internetowych znajduje się duża liczba informacji również na temat obiektów zabytkowych, w tym także wiele fotografii. Problemami są przede wszystkim: ogromna liczba treści, brak uporządkowania oraz trudność w weryfikacji jakości informacji. Istnieją jednak pewne zalety. Za najistotniejszą z nich można uznać aktualność dużej części informacji. Wielu użytkowników Internetu publikuje na stronach WWW prywatne zdjęcia na przykład z wyjazdów wakacyjnych. Na tych fotografiach często znajdują się zabytki (nierzadko stanowiące cel wycieczki). Aktualne fotografie stanowią cenne źródło informacji o stanie poszczególnych obiektów, wykonywanych pracach konserwatorskich, i odbudowach. Takie informacje mogą umożliwić selekcję interesujących fotografii. Tysiące turystów, miłośników zabytków, pasjonatów historii i kultury generuje ogromne ilości nieustannie aktualizowanych treści multimedialnych, których wykorzystanie może być potencjalnie bardzo użyteczne. Tak postawiony problem stwarza jednak wiele istotnych trudności, w tym między innymi: jak określić, które fotografie mogą być przydatne (zawierają przede wszystkim zdjęcia budowli) oraz jak ocenić aktualność fotografii (nie opierając się na jej opisie).

Prezentowana w niniejszym artykule aplikacja umożliwia znajdowanie zdjęć oraz stron internetowych związanych z fotografiami z wykorzystaniem wyszukiwarek internetowych. W celu przeprowadzenia wyszukiwania należy podać konkretną frazę. Aplikacja przeprowa-

dza indeksowanie stron i zapisuje dane (treści oraz obrazy) w bazie danych. Następnie wykonywane jest przeszukiwanie bazy danych i zwrócenie wyników użytkownikowi.

2. Elementy programu wyszukującego

2.1. Web Crawler

Celem działania Web Crawlera jest pobranie z Internetu do własnej bazy danych jak największej liczby dokumentów, które mogą zawierać interesujące nas treści [6].

Web Crawlery można podzielić na trzy grupy wedle sposobu ich działania:

- błądzące – na początku „pająk" posługuje się małą listą stron WWW, które zamierza odwiedzić, następnie przez znalezione na stronach adresy wędruje dalej i poznaje nowe strony;
- posługujące się gotowymi repozytoriami danych – przeszukuje bazy, wyciągając z nich odpowiednie informacje. Wie, które i jak je przeszukiwać;
- mieszane – korzystają z połączenia dwóch wcześniejszych metod. Najpierw odbywa się odnajdywanie potrzebnych informacji z baz danych, a następnie przeszukiwanie ich pod kątem adresów do innych stron WWW.

2.2. Indekser

Indekser to jeden z najważniejszych elementów wyszukiwarki, odpowiedzialny za indeksowanie dokumentów znalezionych przez Web Crawlera. Jest to proces tworzenia bazy danych zawierającej skompilowaną wersję dokumentów ściągniętych przez „pająka" [9]. Baza powinna być zoptymalizowana w celu szybkiego wyszukania listy dokumentów zawierających określone słowa bądź frazy [2]. Proces indeksowania składa się z kilku etapów [1]:

- Identyfikacja słów i fraz znajdujących się w dokumencie.
- Usuwanie słów popularnych.
- Ekstrakcja tematów słów przy użyciu algorytmu szukającego tematu.
- Zastąpienie tematów przez numeryczne identyfikatory termów indeksujących.
- Zliczanie wystąpień tematów.
- Obliczenie wag dla prostych termów, fraz.
- Przypisanie każdemu dokumentowi przynależnych prostych termów, fraz z odpowiednimi wagami.

Przed umieszczeniem w bazie dokument musi zostać odpowiednio przetworzony. Na samym początku należy dokument oczyścić (usunąć kod HTML). Później jest on poddawany procesowi tokenizacji – dzielenia tekstu na słowa. Te najbardziej pospolite (ang. *stop-words*)

są zazwyczaj z tekstu usuwane. Lista pospolitych wyrazów różni się w zależności od języka, a także od rodzaju dokumentów, w jakich ma zostać zastosowana.

Kolejnym etapem jest przeprowadzenie analizy morfologicznej – jest to szczególnie istotny etap dla języków z rozbudowaną fleksją [8]. Wyrazy w języku naturalnym mogą mieć różną formę w zależności od kontekstu; aby umożliwić powiązanie wyrazów z danym leksemem, trzeba w jakiś sposób charakteryzować klasy wyrazów.

Wygenerowane termy są analizowane pod kątem liczby wystąpień w dokumencie, miejsca ich wystąpienia oraz ważności termu na tle całego dokumentu [7]. Dzięki takiej rozbudowanej analizie możliwe jest opracowanie streszczenia badanej strony.

Do indeksowania stron w prezentowanym rozwiązaniu użyto mechanizmów serwera baz danych Oracle.

2.3. Page Rank

Ważną kwestią wyszukiwarek jest ranking stron. Proste zapytania po wcześniejszej analizie zwrócą sporo wyników. By jeszcze lepiej uściślić wybór stron, można wykorzystać strategie szeregowania stron, zwaną inaczej rankingiem stron (ang. Page Rank) [2]. W opisywanym systemie strony szeregowano na podstawie liczby linków na nie wskazujących. Wzór obliczenia Page Rank dla strony A to [1]:

$$PR(A) = (1-d) + d\left(\frac{PR(t_1)}{C(t_1)} + \cdots + \frac{PR(t_n)}{C(t_n)}\right), \tag{1}$$

gdzie: d – współczynnik tłumienia zazwyczaj ustawiony na 0,85; $t_1 \ldots t_n$ – strony zawierające linki do naszej strony; $C(t)$ – liczba linków, do których odsyła dana strona internetowa t.

Tak więc im więcej wartościowych stron o jak najmniejszej liczbie linków wychodzących odwołuje się do naszej strony, tym większa jest wartość Page Rank naszej strony.

2.4. Bazy danych

W programie wykorzystano system bazodanowy Oracle. W skrócie przedstawione zostaną podstawowe mechanizmy użyte do uzyskiwania informacji z bazy. Do przeszukiwania informacji zaimplementowano dwie bazy danych. Schemat został zaczerpnięty z darmowego projektu OpenWebSpider (pierwotnie stworzonego w systemie MySQL) [20].

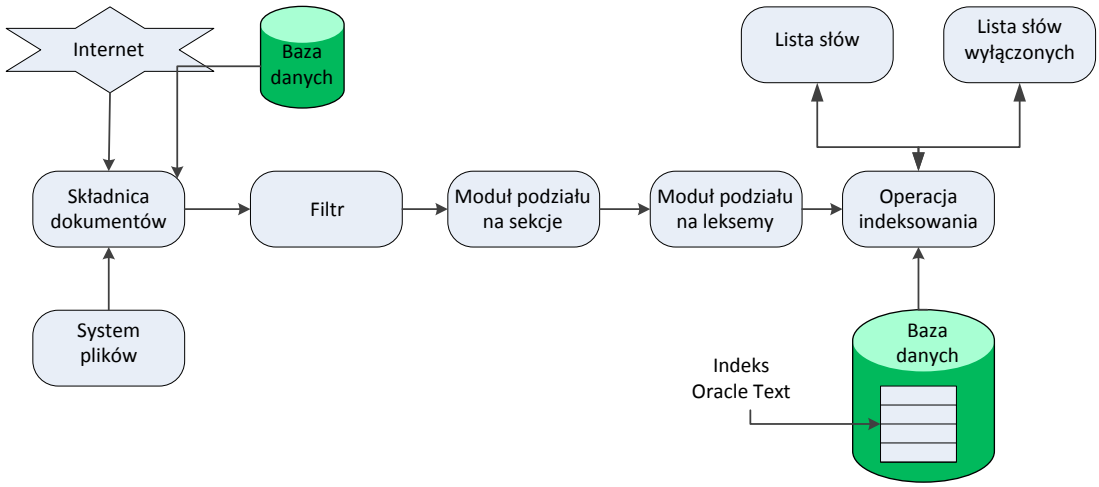
W pierwszej bazie zawarte są informacje dotyczące Web Crawlera oraz adresów do przeszukania. Druga zawiera zindeksowane strony i obrazy.

2.5. Oracle Text

Oracle Text to zintegrowana z serwerem baz danych Oracle technologia wyszukiwania pełnotekstowego (*full-text*). Zapewnia indeksowanie słów i szybkie przeszukiwanie tekstu umieszczonego w bazie danych [4]. Używa standardowego języka SQL do indeksowania, przeszukiwania i analizy tekstu dokumentów.

Pakiet Oracle Text, mimo względnej prostoty użycia, jest technologicznie bardzo złożonym rozwiązaniem. Obejmuje on wszystkie etapy wyszukiwania danych, od określenia źródła danych aż po proces wizualizacji odnalezionych informacji, które spełniają podane przez użytkownika kryteria poszukiwania. Warty podkreślenia jest tutaj fakt, że Oracle Text potrafi przeszukiwać nie tylko dane zgromadzone w systemie Oracle, lecz także doskonale radzi sobie z plikami zewnętrznymi (duży nacisk kładziony jest na format XML), a nawet z plikami „odległymi", dostępnymi za pośrednictwem protokołów HTTP oraz FTP [4]. Dzięki temu zasięg działania Oracle Text może wykraczać daleko poza dane zgromadzone tylko w lokalnych systemach bazodanowych.

Istotą działania modułu Oracle Text jest kilkuetapowa „obróbka" dokumentów wejściowych. Dokument przechodzi przez kolejne moduły, mamy więc do czynienia ze swego rodzaju przetwarzaniem potokowym. Każdy z etapów (modułów) może być parametryzowany przez wykorzystanie określonego zbioru tzw. preferencji. Można więc w sposób jawny wskazać modułowi Oracle Text, że naszym celem jest przykładowo przeszukiwanie dokumentów utworzonych w języku polskim i przechowywanych w bazie w postaci plików HTML. Można również podać, że przeszukiwane będą dokumenty zapisane w formatach doc, pdf oraz xls (Oracle Text automatycznie rozpoznaje ponad 200 różnych formatów) [4].



Rys. 1. Proces indeksowania dokumentów z wykorzystaniem Oracle Text
Fig. 1. Documents indexing process with Oracle Text

Proces indeksowania z wykorzystaniem Oracle Text (rys. 1) obejmuje następujące komponenty [17]:

a) Repozytorim danych. Proces tworzenia indeksu rozpoczyna się od uzyskania danych do przetworzenia z magazynu danych – dopuszcza się trzy możliwości pobierania danych: dokumenty zapisane w bazie danych Oracle, w plikach dyskowych oraz w sieci Internet. Poszczególne dokumenty są pobierane ze wskazanych źródeł i przekazywane do dalszego przetwarzania.
b) Filtr. Po uzyskaniu danych następuje ich filtracja, czyli proces ujednolicenia otrzymywanego formatu danych oraz kodowania. Moduł podaje, w jaki sposób dokumenty powinny być konwertowane do postaci tekstowej. Pliki tekstowe, HTML lub XML nie wymagają filtracji formatu. Pliki binarne, takie jak PDF, muszą zostać przefiltrowane do tekstu ze znacznikami, przy jednoczesnym zachowaniu możliwości wyróżnienia ważniejszych fraz w tekście.
c) Moduł podziału na sekcje (klasyfikowanie tekstu). Polega na rozdzieleniu znaczników od tekstu. Znaczniki służą do obliczania wag danych fraz lub tekstu i są uwzględniane w procesie indeksowania.
d) Moduł podziału na leksemy (lekser). Podaje, w jakim języku są dokumenty, które będą indeksowane. Dzięki tej wiedzy możliwe jest bardziej precyzyjne wydzielenie poszczególnych tokenów (system wówczas wie na przykład, że w języku polskim separatorem dziesiętnym jest przecinek, a w języku angielskim – kropka). W lekserze następuje proces analizy leksykalnej tekstu, który konwertuje słowa występujące w tekście na tokeny, czyli wyrazy mające określone znaczenie.
e) Operacja indeksowania. Silnik indeksujący (ang. *Indexing Engine*) odpowiada za utworzenie odwrotnego indeksu, który dopasowuje słowa do danego dokumentu. Na tym etapie można zdefiniować słowa stopu (lista słów wyłącznych), które mają zostać ignorowane w procesie budowania indeksu (nie będą wpływać na proces wyszukiwania). Dodatkowo można zdefiniować listę słów wykorzystywaną przy procesie tworzenia indeksów prefiksowych lub indeksów zawierających daną frazę.
f) Po przetworzeniu powstaje indeks słów kluczowych, wykorzystywany podczas przeszukiwania.

Tekst (źródło strony) przechowywany jest w bazie danych. Forma tekstu to CLOB (ang. *Character Large Object*).

Istnieją cztery typy indeksów:

- CONTEXT – używany w przypadku dużych dokumentów tekstowych, może indeksować pliki tekstowe i binarne w różnych formatach (np. doc, pdf, ps) [3],
- CTXCAT – używany w przypadku małych dokumentów, indeksuje kolumny CHAR, VARCHAR,
- CTXRULE – to specjalny rodzaj indeksu używanego do budowy aplikacji klasyfikującej dokumenty,

- CTXXPATH – przeznaczonydo przetwarzania dokumentów XML. Można go tworzyć tylko na kolumnach typu XMLType.

3. Klasyfikacja zdjęć

Istotnym elementem prezentowanego podejścia jest zagadnienie rozpoznawania obrazów, umożliwiające rozpoznawanie przynależności obiektów (w tym wypadku zdjęć) do pewnych klas (w pracy istnieją dwie klasy: klasa zabytków oraz klasa pozostałych fotografii różnych od zabytkowych budowli) [16]. Początkowym elementem algorytmu rozpoznającego jest pomiar cech związanych z obiektami. Obiekty te zostają zapisane w postaci wektorów cech. [16]. W niniejszym artykule cechy przyjmują wartości liczb rzeczywistych. System jest uczony i testowany na podstawie przykładów zgromadzonych zdjęć. Dla każdego zdjęcia wyznaczane są wektor powiązanych z nim cech oraz informacja o klasie przynależności [11]. Na potrzeby projektu zgromadzono zbiór 784 zdjęć. Wszystkie metody obróbki obrazów wykorzystane w aplikacji pracują na obrazach wykonanych w skali szarości. System nie analizuje obrazu pod względem kolorów. W zbiorze zdjęć wyodrębniono dwie kategorie: zdjęcia zawierające budowlę oraz zdjęcia nieprzedstawiające budowli. W kategorii zdjęć nieprzedstawiających budowli znalazły się przede wszystkim te, które nie zawierają budynków o charakterze zabytkowym, a zostały wykonane w plenerze (zdjęcia typu *outdoor*). Zdjęcia takie zawierają często fragmenty podobne do tych występujących na ujęciach związanych z budowlami, takie jak np. niebo w górnej części, ale zwykle różnią się znacznie w obszarze środkowym, gdzie mogą wystąpić elementy pejzażu (góry, morze, lasy itp.). Kolejną grupą zdjęć nieprzedstawiających budynków są zdjęcia wykonane wewnątrz pomieszczeń (typu *indoor*), które różnią się już znacznie od zdjęć zawierających budowle, szczególnie w części górnej zdjęcia, gdzie trudno się doszukać np. fragmentów nieba. W zbiorze zdjęć nieprzedstawiających budowli znalazły się również zdjęcia zawierające sylwetki ludzi w różnym stopniu zbliżenia – na niektórych możliwe było rozróżnienie twarzy, co sugerowało, że na pierwszym planie jest raczej osoba, a nie budowla. Wzięto pod uwagę następujące cechy umożliwiające określenie kategorii zdjęcia:

a) średnia jasność pikseli w odpowiednich podobszarach. Zdjęcie podzielone jest na trzy równe części, w których liczona jest średnia wartość jasności (rys. 2). Na zdjęciu przedstawiającym dokumenty, plakaty, obrazy, na całym obszarze jasność rozłożona jest podobnie. Zdjęcia budynków są jasne, przy czym zwykle najjaśniejszy jest górny obszar zdjęcia;
b) zróżnicowanie jasności w podobszarach. Miara zróżnicowania jasności wyrażana jest jako odchylenie standardowe jasności pikseli. Jeżeli górny obszar zdjęcia zawiera obraz

nieba, zróżnicowanie jasności będzie tam niskie, ponieważ obszar ten zawiera piksele o podobnej jasności. Odchylenie standardowe jest liczone w każdej z trzech części obrazu (rys. 5);

c) liczba narożników. Liczone są narożniki znalezione na obrazie. Budowle charakteryzują się dużą liczbą narożników;

Rys. 2. Podział zdjęcia przy wyznaczaniu średniej jasności oraz odchylenia standardowego
Fig. 2. Photo Division during the process of determining the average brightness and standard deviation

d) obecność twarzy. Zdjęcia zabytków, na których znajduje się duża liczba twarzy (lub zdjęcie jest portretem), są bezużyteczne. Użytkownik powinien otrzymać zdjęcie, na którym będzie dobrze uwidoczniony zabytek;
e) liczba linii. Zdjęcia zabytkowych budowli charakteryzują się występowaniem dużej liczby linii (odcinków prostych). Zdjęcia różnią się pod tym względem od portretów, krajobrazów, co ułatwia ich klasyfikację;
f) rozdzielczość zdjęcia. Zdjęcia o małej rozdzielczości nie będą brane pod uwagę.

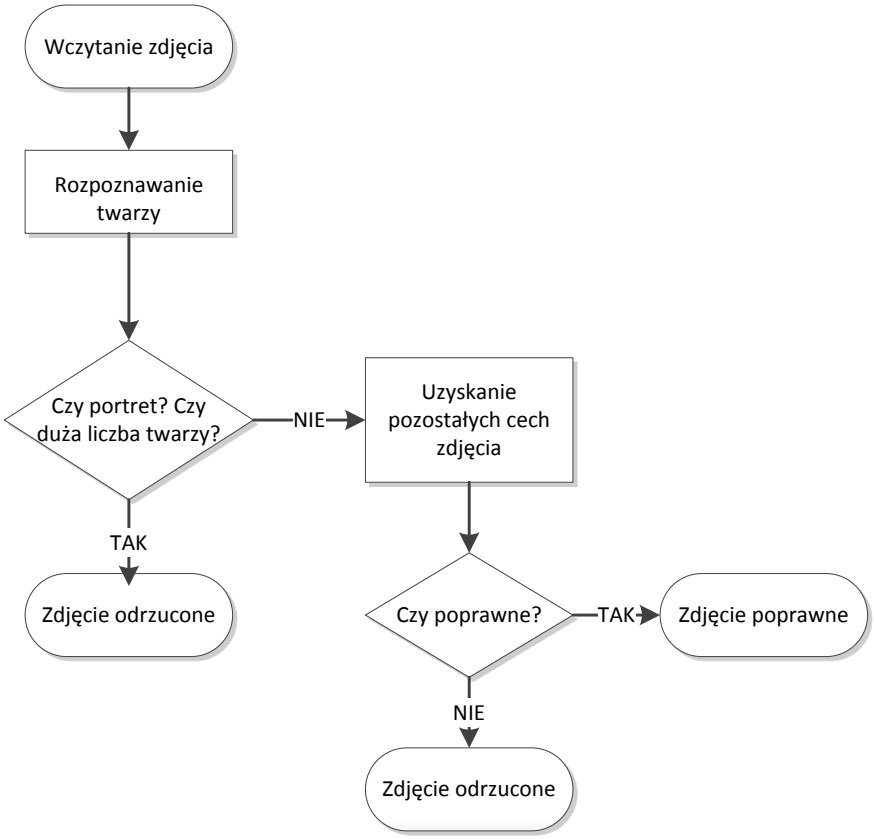
System klasyfikujący biorący pod uwagę wymienione cechy jest oparty na nieliniowej sieci neuronowej, uczonej algorytmem wstecznej propagacji błędów.

Podział obrazu na trzy części był podyktowany spostrzeżeniami popartymi wcześniejszymi doświadczeniami w rozróżnianiu kategorii zdjęć typu *outdoor*, *indoor*, pejzaży, portretów i zdjęć wykonanych w zabudowie miejskiej, dotyczącymi występowania różnej liczby charakterystycznych szczegółów w poszczególnych częściach obrazu (górnej, środkowej i dolnej) w zależności od kategorii.

W celu usprawnienia działania sieci neuronowej każdy obraz jest badany w trzech rozdzielczościach. Jest on przeskalowany, tak aby dłuższa krawędź miała długość kolejno 200, 400, 800 pikseli [19]. Analiza obrazu w trzech różnych skalach była podyktowana głównie celem uzyskania lepszych efektów z analizy wystąpienia odcinków prostych na obrazie. Za-

obserwowano, że przy dużej rozdzielczości na różnych obrazach może wystąpić wiele odcinków prostych, które jednakże nie są miarodajne w kontekście rozróżniania budowli, a wynikają z niemożności precyzyjnego dobrania parametrów dla transformaty Hougha (a w szczególności z faktu, że detektor Canny zwraca zawsze krawędzie o grubości jednego piksela). Dla obrazów pomniejszonych mniej istotne szczegóły ulegały zatarciu, a co za tym idzie mniej było na nich krawędzi i znalezione na ich podstawie proste były bardziej wiarygodne w sensie odzwierciedlania np. struktury budowli, a nie np. pojedynczych liści na drzewie.

Diagram działania algorytmu klasyfikacji zdjęć przedstawiono na rysunku 3.

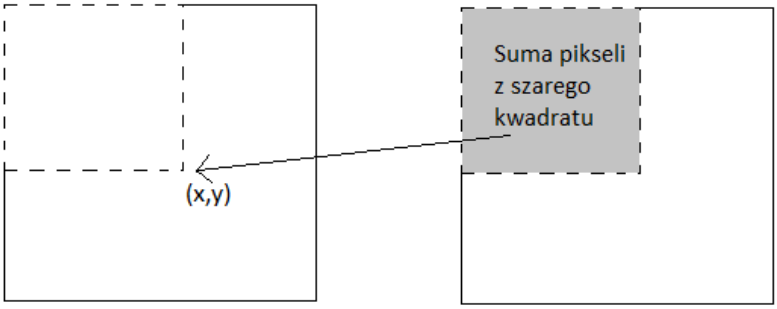


Rys. 3. Algorytm klasyfikacji zdjęć
Fig. 3. Photo classification algorithm

Ostatecznie wektor cech liczy dziesięć liczb: liczbę linii dla obrazu o najdłuższej krawędzi 200, 400 oraz 800, odchylenie standardowe jasności z trzech części obrazu o najdłuższej krawędzi 800 (pierwsza liczba to wartość obliczona dla górnego obszaru, druga dla środkowego, trzecia dla dolnego), liczbę narożników obrazu o najdłuższej krawędzi 800, średnią wartość jasności pikseli z trzech części obrazu o najdłuższej krawędzi 800 (podobna kolejność jak w przypadku odchylenia standardowego). Dane pozyskane ze zdjęcia ulegają procesowi normalizacji względem wartości maksymalnej [12].

3.1. Integral Image

Obliczenie parametrów zdjęcia (mediana, średnia jasność, odchylenie standardowe) jest bardzo czasochłonne, dlatego wykorzystano właściwości obrazu pomocniczego, tzw. *integral image*. To forma obrazu, która w punkcie współrzędnych (x,y) przechowuje wartość równą sumie wartości pikseli znajdujących się na lewo i powyżej punktu (x,y).



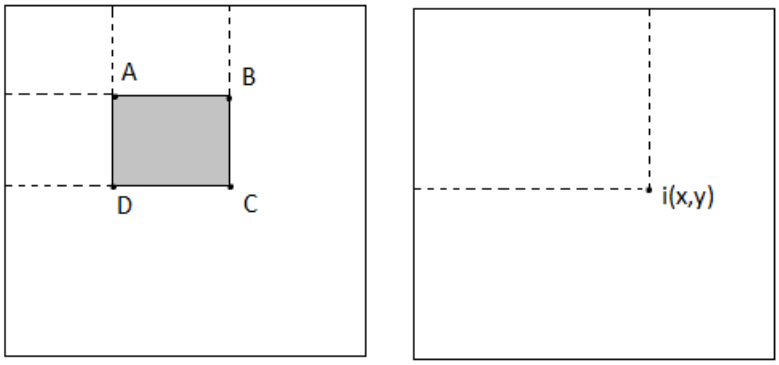
Rys. 4. Obraz typu *integral image*, obraz oryginalny
Fig. 4. Integral image, original image

Przy użyciu tej metody suma jasności pikseli w dowolnym prostokątnym obszarze może być obliczona za pomocą czterech operacji odwołania do pamięci i trzech operacji sumowania. Gdy dysponuje się raz utworzonym *integral image*, w bardzo efektywny sposób można otrzymać sumę wartości jasności pikseli w dowolnym prostokątnym obszarze obrazu wejściowego (rys. 2b – obszar zaciemniony) zgodnie ze wzorem:

$$\sum_{A(x)<x\le C(x),A(y)<y\le C(y)} g(x,y) = i(C)+i(A)-i(B)-i(D),$$

$$D = integralImage(4) - integralImage(3) - integralImage(2) + integralImage(1) \qquad (2)$$

gdzie $g(x, y)$ jest jasnością piksela na obrazie w skali szarości.



Rys. 5. Sposób obliczania wartości *integral image*
Fig. 5. Calculation of integral image

3.2. Rozpoznawanie twarzy

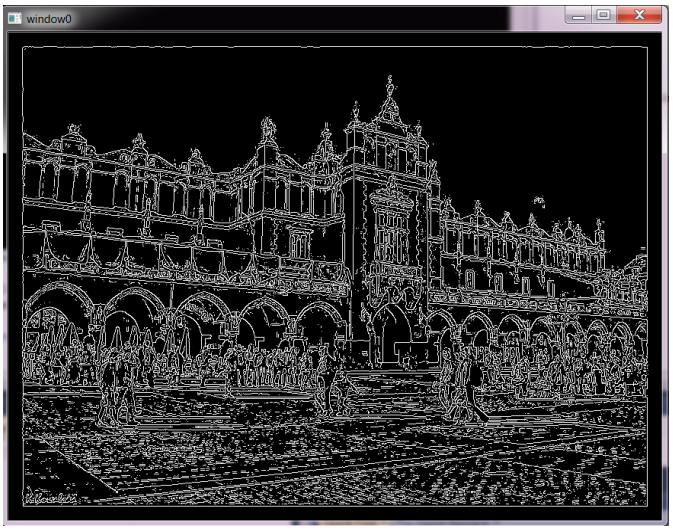
W celu wykrycia twarzy wykorzystano metodę zwaną detektorem Viola-Jones. Metoda ta została stworzona przez Paula Violę i Michaela Jonesa. Celem ich pracy było opracowanie

skutecznego systemu wykrywania twarzy, który miał zapewniać dużą szybkość działania. Metoda polega na przechodzeniu po obrazie wejściowym oknem detektora, w którym badane jest działanie zestawu klasyfikatorów [16].

Detektor bazuje na cechach typu Haara, które można szybko obliczyć za pomocą *integral image*. Cechy te wykorzystywane są przy budowie kaskady klasyfikatorów trenowanych za pomocą algorytmu AdaBoost. Zadaniem takiego klasyfikatora jest sprawdzenie zadanych obszarów obrazu i podjęcie decyzji, czy dany obraz będzie zakwalifikowany do jednej z dwóch klas (klasa „twarzy" albo „nie twarzy"). W aplikacji wykorzystano wytrenowaną kaskadę dostępną za pomocą biblioteki OpenCV.

3.3. Wykrywanie linii

W celu znalezienia linii wykorzystano transformatę Hougha. Oblicza się ją zwykle na obrazie czarno-białym zawierającym punkty należące do krawędzi. W celu ich uzyskania zastosowano metodę Canny'ego. Każdy punkt obrazu może należeć do jakiegoś zbioru prostych, które mogą przez niego przechodzić. Jeśli prostą opiszemy przez współczynnik jej pochylenia a (dy/dx) i przesunięcia b, to punkt obrazu oryginalnego (y=ax+b) transformowany jest do nowego położenia w przestrzeni określonej parametrami (a, b) i związanego ze wszystkimi liniami przez niego przechodzącymi. Jeśli każdemu rozpatrywanemu (pod kątem szukania linii) punktowi obrazu wejściowego (punktom białym z obrazu po detekcji krawędzi, rys. 6) przypisalibyśmy pęk prostych przez niego przechodzących, a co za tym idzie – zbiór punktów w przestrzeni Hougha (a,b), to linie występujące w obrazie oryginalnym byłyby powiązane z miejscami, gdzie występują maksima w przestrzeni (a,b).



Rys. 6. Rezultat działania algorytmu Canny'ego
Fig. 6. Results of Canny's Algorithm

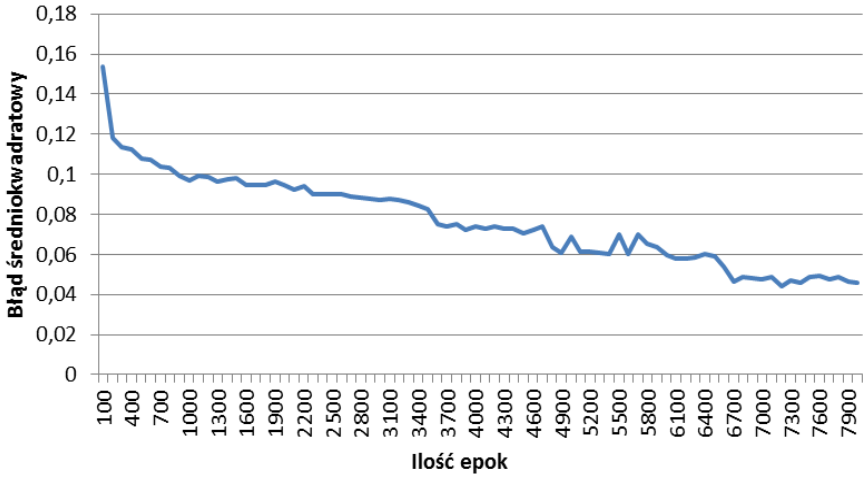
Wybór parametrów a i b do określenia nowej przestrzeni nie jest jednak najlepszym rozwiązaniem, głównie ze względu na nieskończoną wartość a w przypadku linii pionowych.

Z tego też względu stosuje się zazwyczaj zapis parametrów linii we współrzędnych polarnych: $r = x_0 cos\theta + y_0 sin\theta$, gdzie r jest odległością linii od początku układu współrzędnych (pod kątem prostym), a θ jest kątem, jaki tworzy linia prostopadła do danej a osią x.

3.4. Sieć neuronowa

Zastosowana w programie sieć to sieć nieliniowa z sigmoidalną funkcją aktywacji neuronów. Sieć składa się z warstwy wejściowej (osiem neuronów), dwóch warstw ukrytych (pierwsza warstwa ukryta liczy czterdzieści neuronów, druga – dwadzieścia) oraz warstwy wyjściowej (dwa neurony). Parametry sieci (w szczególności liczbę warstw ukrytych oraz ilość występujących w nich neuronów) dobrano na podstawie licznych testów, które przeprowadzono dla szerokiego spektrum tych parametrów. Uzyskana konfiguracja ma zapewnić przede wszystkim dobrą zdolność generalizacji systemu w procesie klasyfikacji, co jest szczególnie ważne w kontekście dużej różnorodności zdjęć, jakie możemy napotkać w praktyce.

W procesie nauki wykorzystano 392 pozytywne zdjęcia oraz 392 negatywne. W celu analizy nauki sieci neuronowej zastosowano miarę błędu średniokwadratowego (ang. *Mean Square Error* – MSE) [11]. Błąd był mierzony co sto epok. Przykładowy przebieg procesu uczenia przedstawia wykres na rysunku 7. W aplikacji wykorzystano parametry sieci: momentum – 0,4, współczynnik uczenia – 0,2 (dobrane na podstawie testów).



Rys. 7. Wykres błędu w trakcie przebiegu uczenia sieci neuronowej
Fig. 7. Mean Square Error during neural network training

4. Klasyfikacja stron

Niejednokrotnie można natknąć się na sytuację, w której wyszukiwanie po zadanym słowie kluczowym zwróci wyniki dla różnych znaczeń danego słowa. Wyszukiwanie według

słowa „koloseum” może zwrócić zarówno wyniki związane z obiektem zabytkowym, jak i dla lokalu gastronomicznego. W rozwiązaniu problemu niejednoznaczności może pomóc wcześniejsza klasyfikacja stron WWW ze względu na ich kategorię.

Aplikacja zawiera możliwość wyboru kategorii stron, jakie chcemy indeksować.

Kategorie, które brane są pod uwagę, to:

- Artykuł – strony charakteryzujące się dużą ilością treści, mało zróżnicowane jeśli chodzi o wygląd. Przykłady: artykuły naukowe, recenzje, reportaże itp.
- Blog – stosunkowo nowy gatunek stron, charakteryzujący się podobną formą (chronologiczna lista wpisów, dużo indywidualnych zwrotów).
- E-sklep – strony charakteryzujące się dużą ilością cen, listami artykułów, linków.
- FAQ – zbiór stron pomocy, charakteryzujących się dużą liczbą pytań i stosunkowo krótkich odpowiedzi.
- Forum – strony charakteryzujące się dużą liczbą wypowiedzi, o charakterystycznej formie (awatary, powtarzające się formatowanie postów).
- Katalog – zbiór dużej liczby linków internetowych.
- Portal – strony portali łączące w sobie dużo elementów multimedialnych, wyszukiwarki, linki oraz krótkie artykuły.
- Strona domowa – strony o prostej formie, zazwyczaj prezentujące osobę, zawierają raczej nieformalną treść.
- Strona firmowa – strona reprezentująca firmy bądź instytucje, zawiera dużo treści oraz linków. Strony charakteryzują się podobnym formalnym językiem oraz słowami kluczowymi (firma, kontakt, kariera itp.).

W programie istnieje możliwość wykorzystania dwóch metod klasyfikacji. Jedna to metoda oparta na algorytmie AdaBoost [15, 16], druga wykorzystuje zastosowanie zbiorów przybliżonych [18].

5. Działanie programu i wyniki testów

W celu przeprowadzenia wyszukiwania należy podać konkretną frazę (np. „kościół mariacki”). Aplikacja rozpocznie przeszukiwanie baz danych. Wszelkie dane, które zostaną zaprezentowane, wybrane są z wcześniej przygotowanej bazy danych. Jeśli informacje na temat danego zabytku nie znajdują się w bazie danych, wyszukiwarka poinformuje użytkownika

o braku danych. Aby zebrać dane, należy skorzystać z programu indeksującego, który odpowiednio uzupełni bazę danych. Strony internetowe i treści graficzne są przetrzymywane w bazie i poddane są (w momencie wprowadzenia ich do bazy) procesowi indeksowania w celu szybszego wyszukiwania informacji.

Na samym początku należy wybrać odpowiednią metodę: opierającą się na wykorzystaniu Google, Bing, podanym adresie katalogu stron lub przygotowanym pliku z listą stron.

W celu weryfikacji proponowanego rozwiązania przeprowadzono testy mające sprawdzić skuteczność zastosowanego podejścia do problemu wyszukiwania i analizy danych internetowych na temat wybranych aspektów obiektów zabytkowych. Eksperyment polegał na sprawdzeniu, jak skutecznie sieć neuronowa kategoryzuje fotografie oraz czy klasyfikacja stron internetowych może wpłynąć na poprawność procesu wyszukiwania informacji.

5.1. Testy programu

W każdym z trzech poniższych testów wykorzystano wcześniej przygotowane adresy URL stron związanych z nazwą odpowiedniego zabytku. Do każdej kategorii stron przyporządkowano pewną liczbę adresów URL. Test pierwszy przeszukuje dane adresy URL w celu odnalezienia informacji na temat opery we Lwowie, drugi – na temat kościoła Mariackiego, trzeci – na temat pałacu Esterhazyego.

W teście pierwszym wykorzystano 73 adresy URL. Strony użyte w teście zawierały między innymi informację na temat opery we Lwowie (która była zabytkiem poszukiwanym).

Tabela 1

Wyniki działania sieci neuronowej w teście pierwszym

Liczba znalezionych zdjęć	Liczba zdjęć przedstawiających budowle	Liczba zdjęć nieprzedstawiających zabytków	Poprawne rozpoznania przez sieć neuronową
437	315	122	73%

Tabela 2

Wyniki przeszukiwania stron o zadanych kategoriach w teście pierwszym

Typ strony	L. stron	L. zdjęć pobranych ze stron	L. zdjęć przedstawiających budowlę	L. zdjęć nieprzedstawiających zabytków	L. stron przedstawiających szukany zabytek
Portal	12	28	24	4	3
S. domowa	14	128	91	37	10
S. firmowa	9	74	48	26	2
Artykuł	15	41	33	8	0
Forum	6	140	108	32	5
E-sklep	4	1	1	0	0
Blog	10	23	9	14	7
Katalog	2	1	0	1	0
FAQ	1	1	1	0	0

Z testu wynika, że większość zdjęć przedstawiających szukany zabytek znajduje się na stronach typu strona domowa, forum internetowe oraz blog (tabela 2). Wykorzystując tę wiedzę, można skrócić czas przeszukiwania Internetu przez wykluczenie pozostałych typów stron.

Do testu drugiego wykorzystano 78 stron. Strony wykorzystane w teście zawierały informację na temat kościoła Mariackiego.

Tabela 3

Wyniki działania sieci neuronowej w teście drugim

Liczba znalezionych zdjęć	Liczba zdjęć przedstawiających budowle	Liczba zdjęć nieprzedstawiających zabytków	Poprawne rozpoznania przez sieć neuronową
143	121	22	85%

Podobnie jak w przypadku testu pierwszego, w teście drugim stwierdzono, że większość zdjęć przedstawiających szukany zabytek znajduje się na stronach typu e-sklep, strona domowa, oraz blog (tabela 4).

Tabela 4

Wyniki przeszukiwania stron o zadanych kategoriach w teście drugim

Typ strony	L. stron	L. zdjęć pobranych ze stron	L. zdjęć przedstawiających budowlę	L. zdjęć nieprzedstawiających zabytków	L. stron przedstawiających szukany zabytek
Portal	13	38	25	13	2
S. domowa	15	36	33	3	10
S. firmowa	16	32	28	4	4
Artykuł	11	10	10	0	2
Forum	4	3	2	1	1
E-sklep	7	4	4	0	3
Blog	7	20	19	1	5
Katalog	4	0	0	0	0
FAQ	1	0	0	0	0

Do testu trzeciego również wykorzystano 78 stron. Strony wykorzystane w teście zawierały informację na temat pałacu Esterhazyego.

Tabela 5

Wyniki działania sieci neuronowej w teście drugim

Liczba znalezionych zdjęć	Liczba zdjęć przedstawiających budowle	Liczba zdjęć nieprzedstawiających zabytków	Poprawne rozpoznania przez sieć neuronową
340	293	97	71%

Również na podstawie testu trzeciego stwierdzono, że większość zdjęć przedstawiających szukany zabytek znajduje się na stronach typu portal, blog, e-sklep, strona domowa, forum internetowe (tabela 6).

Wiele stron typu katalog, portal, FAQ, e-sklep, strona firmowa porusza tylko kwestie dotyczące zabytków, nie zawierając ich zdjęć. Strony typu blog, forum internetowe, strony domowe zawierają często zdjęcia zabytków i zwykle są to zdjęcia prezentujące aktualny stan tych obiektów. Wprowadzenie indeksowania z możliwością wyboru kategorii przeszukiwa-

nych stron pozwala zaoszczędzić czas przez ominięcie indeksowania stron, które często nie zawierają istotnych z naszego punktu widzenia treści (aktualnych zdjęć obiektów zabytkowych) oraz wpływa na precyzję otrzymanych wyników.

Tabela 6

Wyniki przeszukiwania stron o zadanych kategoriach w teście drugim

Typ strony	**L. stron**	**L. zdjęć pobranych ze stron**	**L. zdjęć przedstawiających budowlę**	**L. zdjęć nieprzedstawiających zabytków**	**L. stron przedstawiających szukany zabytek**
Portal	5	5	4	1	2
S. domowa	10	15	8	7	4
S. firmowa	14	42	37	5	2
Artykuł	10	0	0	0	0
Forum	16	79	56	23	7
E-sklep	5	3	3	0	2
Blog	9	195	134	61	7
Katalog	1	1	1	0	0
FAQ	0	0	0	0	0

6. Podsumowanie

W artykule przedstawiono przykład autorskiego rozwiązania umożliwiającego polepszenie wyszukiwania danych tekstowych i graficznych w sieci Internet dotyczących obiektów zabytkowych. Opracowywana aplikacja może stanowić wsparcie w pracach konserwatorskich.

W artykule scharakteryzowano zagadnienia związane z sieciami neuronowymi oraz rozpoznawaniem obrazów, wykorzystane w opracowywanej aplikacji. Omówiono metody obróbki obrazu, takie jak detektor Viola-Jones czy wykrywanie linii za pomocą standardowej transformaty Hougha. Algorytmy posłużyły do pozyskania różnych cech ze zdjęcia, umożliwiających naukę sieci neuronowej dokonującej klasyfikacji zdjęć. Ponadto skupiono się na zagadnieniach przeszukiwania Internetu, indeksowania stron oraz składowania informacji. W artykule wykazano, iż klasyfikacja stron internetowych ze względu na wprowadzone kategorie wykorzystana podczas procesu indeksowania znacząco wpływa na jakość uzyskiwanych wyników.

Planowana jest dalsza rozbudowa prezentowanego podejścia przy wzięciu pod uwagę rozwoju modułu związanego z rozpoznawaniem obrazów (np. przez zwiększoną liczbę cech charakteryzujących zdjęcia) oraz modułu odpowiedzialnego za kategoryzację stron. W tym ostatnim można uwzględnić zaawansowane cechy wpływające na ranking stron, takie jak popularność (liczbę kwerend zadawanych przez internautów, którym odpowiada dana strona) oraz lokalizację.

BIBLIOGRAFIA

1. Kłopotek M. A.: Inteligentne wyszukiwarki internetowe. Akademicka Oficyna Wydawnicza EXIT, Warszawa 2001.
2. Abiteboul S., Buneman P., Suciu D.: Dane w sieci WWW. Wyd. MIKOM, Warszawa 2003.
3. Searching the Web Effectively. University of Kentucky, HRD Technology Training, 2003.
4. Jansons S.: Getting on the right track with Search Engines. PageWorks, 2003.
5. Archibald T.: Content-Aware Search. Infoworld, 2002, s. 34.
6. Olston C., Najork M.: Web Crawling. USA, Hannover 2010.
7. Chang G.: Mining the World Wide Web. Kluwer Academic Publishers, USA 2001.
8. Articles on Internet Search Algorithms, Including: Web Crawler, Distributed Web Crawling, Proximity Search (Text), Federated Search, URL Normalization, Hilltop Algorithm, Index (Search Engine), Image Meta Search, Web Harvesting, Hephaestus Books 2011.
9. Tan Q.: Designing New Crawling and Indexing Techniques for Web Search Engines, 2008.
10. Tadeusiewicz R.: Sieci neuronowe. AOW RM, Warszawa 1993.
11. Tadeusiewicz R.: Rozpoznawanie obrazów. PWN, Warszawa 1991.
12. Tadeusiewicz R., Gąciarz T., Borowik B., Leper B.: Odkrywanie właściwości sieci neuronowych przy użyciu programów w języku C#. Polska Akademia Umiejętności, Warszawa 2007.
13. Szmygin B.: System ochrony zabytków w Polsce – analiza, diagnoza, propozycje. Polski Komitet Narodowy ICOMOS, 2011.
14. Krajowy Ośrodek Badań i Dokumentacji Zabytków, http://www.kobidz.pl/.
15. Gąciarz T., Czajkowski K., Niebylski M., Szawernoga R.: Klasyfikacja stron internetowych z wykorzystaniem algorytmu boostingu. Studia Informatica, Vol. 32, No. 2A (96), Gliwice 2011.
16. Gąciarz T., Czajkowski K., Niebylski M.: Adaboost ranking results improvement by pairwise classifiers for web page classification, [in:] Czachórski T., Kozielski S., Stańczyk U. (eds.): Advances in Intelligent and Soft Computing, Vol. 103, Man-Machine Interactions 2, Springer-Verlag, Berlin/Heidelberg 2011, s. 393÷400.
17. Czajkowski K., Ziajka D.: Zaawansowane indeksowanie i wyszukiwanie danych w systemie Oracle. Studia Informatica, Vol. 32, No. 2B (97), Gliwice 2011.
18. Czajkowski K.: Reguły decyzyjne i bazy danych w klasyfikacji stron internetowych. Studia Informatica, Vol. 31, No. 2A (89), Gliwice 2010.

19. Gąciarz T., Czajkowski K.: Klasyfikacja zdjęć w multimedialnych bazach danych. Studia Informatica, Vol. 31, No. 2A (89), Gliwice 2010.
20. Open Source Web Spider, http://www.openwebspider.org/.

Wpłynęło do Redakcji 14 stycznia 2013 r.

Abstract

The monuments protection is at present one of the key directions of action which aim is to preserve cultural heritage of particular countries. Diversity of objects which are recognized as monuments and a long time interval of construction make this task challenging. Activities that cover the restoration of objects to their former appearance, often require the reconstruction of the elements that do not exist anymore. Database of monuments can be the support for conservation works. Unfortunately, many countries do not have a central system which collects such information. Existing systems do not possess enough accurate and actual photographic documentation, which is a very valuable source of information.

The Internet, a huge source of information concerning historic buildings, including their photographs. The paper presents the approach that uses the set of features which can be used in photos selection from the point of view of their usefulness. In the paper, issues connected with neural networks and image recognition used in presented application were characterized. In addition, authors focused on the issues of searching the Internet, indexing web pages and data storing.

Adresy

Marek BANACH: Politechnika Krakowska, Wydział Fizyki, Matematyki i Informatyki, Instytut Teleinformatyki, ul. Warszawska 24, 31-155 Kraków, Polska, kaleta.marcin88@gmail.com.
Krzysztof CZAJKOWSKI: Politechnika Krakowska, Wydział Fizyki, Matematyki i Informatyki, Instytut Teleinformatyki, ul. Warszawska 24, 31-155 Kraków, Polska, kc@pk.edu.pl.
Tomasz GĄCIARZ: Politechnika Krakowska, Wydział Fizyki, Matematyki i Informatyki, Instytut Teleinformatyki, ul. Warszawska 24, 31-155 Kraków, Polska, tga@pk.edu.pl.