

Michał MAJEWICZ, Tomasz GAĆIARZ, Krzysztof CZAJKOWSKI
Politechnika Krakowska, Instytut Teleinformatyki, Wydział Fizyki, Matematyki i Informatyki

DETEKCJA REKLAM W BAZACH DANYCH VIDEO

Streszczenie. Reklamy telewizyjne są obecne niemal w każdym emitowanym materiale telewizyjnym. Umieszczanie takich treści w bazach danych video powoduje dodatkowe problemy, z których najważniejsze to większy rozmiar plików oraz dłuższy czas ich przeszukiwania. W artykule przedstawiono metody wykrywania i usuwania niepożądanych treści reklamowych z materiału video, z wykorzystaniem klasyfikatorów bazujących na zestawach tzw. cech typu Haara. Proponowany algorytm, dzięki dostosowaniu do konkretnej stacji telewizyjnej, osiąga wysoki poziom dokładności detekcji.

Słowa kluczowe: bazy danych video, wykrywanie reklam, ekstrakcja cech obrazu, analiza materiału video, cechy Haara

DETECTING OF TV COMMERCIALS IN VIDEO DATABASES

Summary. TV commercials are present in almost each currently emitted telecast. Placing such content in video databases creates additional problems, most important of which is a larger file size and longer duration of their search. This paper shows methods of detecting and removing of undesirable TV commercials from video material by using classifiers based on Haar-like features. The proposed algorithm can adapt to a particular TV channel and achieves a high level of precision.

Keywords: video databases, tv commercials detecting, extracting image features, video material analyzing, Haar-like features

1. Wstęp

Bazy danych video stawiają poważne wyzwania w zakresie pozyskiwania, składowania i przetwarzania danych. Jednak największym problemem jest wyszukiwanie informacji w tego typu bazach. Od wielu lat prowadzone są bardzo liczne prace mające na celu przeszukiwa-

nie treści video w bazach danych [1, 2, 3, 4, 5]. W przypadku rejestrowania sygnału telewizyjnego dodatkowym problemem bez wątpienia są również reklamy, pojawiające się na kanałach komercyjnych z dużą częstotliwością i przerywające nadawanie właściwego materiału. Powoduje to, że duża część danych jest prawdopodobnie nieistotna, stanowi ona jednak również integralną część poszczególnych plików i wymaga tych samych operacji co dane mające wartość dla użytkownika, zwiększając w szczególności czas przeszukiwania. Z uwagi na fakt, że proces przeszukiwania charakteryzuje się często wysoką złożonością czasową i obliczeniową, wyeliminowanie z niego wszelkich zbędnych treści może znacząco poprawić wydajność operacji. Ręczne usuwanie reklam telewizyjnych z plików video, chociaż teoretycznie możliwe, pochłania dużo czasu, nie wspominając o konieczności użycia odpowiednich narzędzi edytorskich. Narzędzia te, chociaż skuteczne, w przeważającej większości są płatne, a ich cena jest niewspółmierna w stosunku do pożądanej w omawianym przypadku funkcjonalności. Z tego powodu odpowiednie wydają się podejścia umożliwiające automatyczną detekcję i eliminację reklam oraz cechujące się wysoką skutecznością i szybkością działania. Zaimplementowane rozwiązanie stanowić może jeden z modułów systemu zarządzania multimedialną bazą danych. Moduł taki uruchamiany będzie dla wszystkich, bądź tylko dla wybranych, plików video (stacje niekomercyjne zwykle nie przerywają nadawania programów blokami reklamowymi), jako jeden z pierwszych etapów w procesie przesyłania plików do bazy danych, zwiększając efektywność funkcjonowania innych elementów systemu.

2. Metody wykrywania reklam telewizyjnych

2.1. Przegląd znanych metod detekcji reklam

Głównym problemem pojawiającym się podczas procesu usuwania reklam z konkretnego pliku video jest opracowanie sposobu odróżniania przez komputer pożądanej przez człowieka treści (tzw. bloki programowe¹) od treści, która ma docelowo zostać usunięta (tzw. bloki reklamowe²). Obecnie znane metody wykrywania reklam telewizyjnych można podzielić na trzy kategorie: metody oparte o śledzenie logo stacji TV, metody oparte o ekstrakcję cech oraz metody korzystające z bazy zaufanych wzorców.

W tym rozdziale omówiono pokrótce każdą z tych kategorii.

¹ Przez *blok programowy* rozumie się ciągłą w czasie sekwencję obrazów, tworzącą niepodzielną całość, zawierającą treści pożądane przez odbiorcę.

² Przez *blok reklamowy* rozumie się ciągłą w czasie sekwencję pojedynczych reklam telewizyjnych, tworzących niepodzielną całość.

2.1.1. Metody oparte na śledzeniu logo stacji TV

Algorytmy należące do tej rodziny wykorzystują fakt, iż w większości światowych kanałów telewizyjnych logo stacji TV jest obecne podczas emisji bloku programowego, podczas gdy nie jest wyświetlane w czasie emisji bloku reklamowego. Znając położenie logo na kolejnych klatkach materiału video, możliwe jest śledzenie jego obecności, co przekłada się na fakt emisji bloku filmowego lub reklamowego.

Duża część algorytmów z tej rodziny bazuje na wykrywaniu krawędzi logo.

W pierwszym kroku algorytm, na podstawie serii kilkudziesięciu do kilkuset klatek, tworzy maskę bitową logo telewizyjnego. W późniejszym etapie stworzona maska używana jest do analizy porównawczej obrazu wyświetlanego w określonym miejscu klatki z otrzymaną maską bitową śledzonego emblematu [20].

2.1.2. Metody oparte na ekstrakcji cech

Algorytmy z tej rodziny opierają się na ekstrakcji zbioru cech charakterystycznych dla bloku programowego oraz dla bloku reklamowego, a następnie opracowaniu na ich podstawie klasyfikatora zdolnego do odróżniania pożądaných fragmentów nagrania od tych niepożądanych przez odbiorcę.

Stosowane tu algorytmy różnią się charakterystyką oraz sposobem ekstrakcji i wykorzystania cech charakterystycznych dla bloków programowych i bloków reklamowych.

2.1.2.1. Wykrywanie czarnych klatek

Czarne klatki (ang. *black frames*) to potocznie używana nazwa określająca serię kilkunastu całkowicie czarnych i jednolitych obrazów wyświetlanych przed emisją bloku reklamowego oraz po jego zakończeniu.

Algorytm detekcji bloków reklamowych, korzystając z tego faktu, przeszukuje sekwencyjnie plik video z nagraniem w poszukiwaniu czarnych klatek – odnalezione miejsca są następnie oznaczane jako początek lub koniec bloku reklamowego.

W celu zwiększenia skuteczności wykrywania, metodę tę często łączy się z wykrywaniem tzw. cichych klatek (ang. *silent frames*), czyli równoczesnym śledzeniem nagłego zaniku sygnału ścieżki dźwiękowej [21].

Czarne klatki są jednak stosowane głównie w krajach Europy Zachodniej, przez co algorytmy oparte na ich wykrywaniu nie znajdują zastosowania w polskich realiach.

2.1.2.2. Badanie poziomu natężenia sygnału dźwiękowego

Powszechną praktyką, stosowaną m.in. w polskich stacjach telewizyjnych, jest gwałtowne zwiększenie poziomu natężenia dźwięku w momencie rozpoczęcia emisji bloku reklamowego. Teoretycznie algorytm opierający się na tym założeniu jest w stanie wykryć początek

i koniec poszczególnych bloków reklamowych, badając natężenie sygnału dźwiękowego w całym pliku video.

Po uchwaleniu ustawy Krajowej Rady Radiofonii i Telewizji (KRRiT) z dn. 15 grudnia 2009 r., zmieniającej rozporządzenie w sprawie sposobu prowadzenia działalności reklamowej i telesprzedaży w programach radiowych i telewizyjnych, podejście to teoretycznie traci rację bytu.

2.1.2.3. Śledzenie niskopoziomowych cech strumienia video/audio

Algorytmy zaliczane do tej kategorii bazują na niskopoziomowych cechach ekstrahowanych ze strumienia video/audio. Cechy te mogą opierać się na badaniu odległości pomiędzy poszczególnymi klatkami kluczowymi (ang. *key frames*), badaniu natężenia luminancji w poszczególnych makroblokach strumienia MPEG czy wykrywaniu gwałtownych zmian dynamiki scen pomiędzy sąsiadującymi klatkami, przez badanie wektorów ruchu [22].

2.1.3. Metody oparte na zaufanych bazach wzorców

Podejście to znacząco różni się od tych opisanych w poprzednich paragrafach. Algorytmy zaliczane do tej rodziny opierają się na stworzonych wcześniej zaufanych bazach danych, przechowujących wzorce znanych reklam telewizyjnych. Eliminacja bloków reklamowych odbywa się poprzez porównywanie poszczególnych fragmentów nagrania ze znanymi reklamami zapisanymi uprzednio w bazie danych. Reklamy przechowywane w bazie są zapisane w postaci tzw. odcisku palca (ang. *fingerprint*), opisanego przez jednoznaczny skrót.

Budowa „odcisków palców” dla poszczególnych reklam bazuje na ich charakterystycznych cechach, jak np. jasność i nasycenie pikseli obrazu w poszczególnych obszarach klatek czy proporcje wyświetlania obrazu.

Z pewnością wadą przedstawionego wyżej rozwiązania jest konieczność ręcznego zarządzania bazą danych przechowującą wzorce znanych reklam, a także brak możliwości wykrywania reklam nieznanymi przez system [23].

2.2. Rozwiązania przyjęte do realizacji

Wnikliwa analiza materiału video, pozyskanego z treści nadawanych przez polskie stacje telewizji komercyjnej, wykazała, że najbardziej stabilną cechą pozwalającą wyróżnić bloki programowe spośród bloków reklamowych jest obecność logo nadawcy. Zasadniczo jest to jedyny element wyświetlany w zdefiniowanym miejscu obrazu (na ogół prawy, górny róg), a dodatkowo jego wygląd zmienia się bardzo rzadko.

Do testów skuteczności, a tym samym użyteczności, obranej metody zdecydowano się na wybór stacji TVN®. Decyzja ta podyktowana jest z jednej strony jednoznacznie określonymi momentami wyświetlania logo w trakcie emisji programu (logo jest wyświetlane jedynie

w trakcie trwania bloku programowego), a zdrugiej relatywnie bardziej skomplikowanym kształtem logotypu w stosunku do pozostałych stacji telewizji komercyjnych.

Bardziej skomplikowany kształt znacząco zwiększa gwarancję sukcesu wykrywania prostszych logotypów.

2.3. Cechy typu Haara

Najbardziej stabilną właściwością pozwalającą odróżnić właściwe treści programowe spośród treści reklamowych jest obecność logo nadawcy. Do jego detekcji użyto tzw. cech Haara [6, 8, 9, 18]. Cechy te charakteryzują się dużą szybkością obliczeń i łatwością implementacji, co ma niebagatelne znaczenie zwłaszcza przy implementacji w systemach wbudowanych.

Pojedyncza cecha typu Haara składa się z minimum dwóch prostokątnych obszarów. Każdy obszar może być oznaczony kolorem białym (w tych miejscach spodziewana jest względnie większa wartość sumy wartości pikseli) albo kolorem czarnym (w tym miejscach spodziewana jest względnie mniejsza jasność sumy wartości pikseli). Warto zaznaczyć, że do obliczeń przyjmuje się, że biały prostokąt wypełnia cały obszar cechy, a prostokąty czarne – jedynie jej fragmenty.

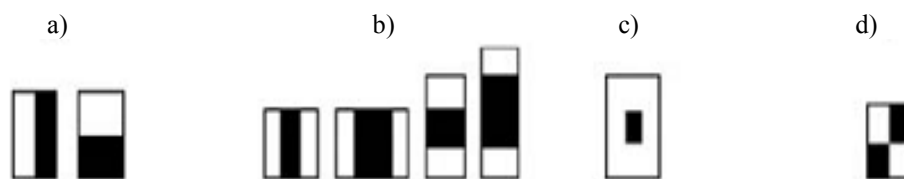
Wartość pojedynczej cechy oblicza się jako różnicę sumy wartości jasności pikseli w białym obszarze oraz sumy wartości jasności pikseli w obszarze czarnym, pomnożonych przez odpowiednie współczynniki korekcyjne. Współczynniki te są zwykle dobierane w taki sposób, aby zrównoważyć pola powierzchni obszarów oznaczonych kolorem czarnym z polem powierzchni oznaczonym kolorem białym.

Wartość pojedynczej cechy Haara opisuje wzór (1):

$$F(x) = W \cdot W_w - B \cdot W_b, \tag{1}$$

gdzie: $F(x)$ – wartość cechy Haara, W – suma wartości jasności pikseli w obszarze oznaczonym kolorem białym, W_w – współczynnik korekcyjny obszaru białego, B – suma pikseli w całkowitym obszarze oznaczonym kolorem czarnym, W_b – współczynnik korekcyjny obszaru czarnego.

Podział podstawowych cech Haara został przedstawiony na rysunku 1.



Rys. 1. Zgrupowane cechy Haara: a) cechy krawędziowe, b) cechy liniowe, c) cechy punktowe, d) cechy diagonalne

Fig. 1. Grouped Haar-like features: a) edge features, b) linear features, c) point features, d) diagonal features

Odczytując wartość cechy Haara nałożonej na dany fragment obrazu graficznego, na podstawie odczytanej wartości $F(x)$ można w łatwy sposób stwierdzić obecność lub brak obecności obiektu w badanym obszarze. Tabela 1 przedstawia zależność jasności badanych obszarów względem siebie, od wartości nałożonej w tym miejscu cechy.

Tabela 1
Zależność jasności badanych obszarów względem siebie

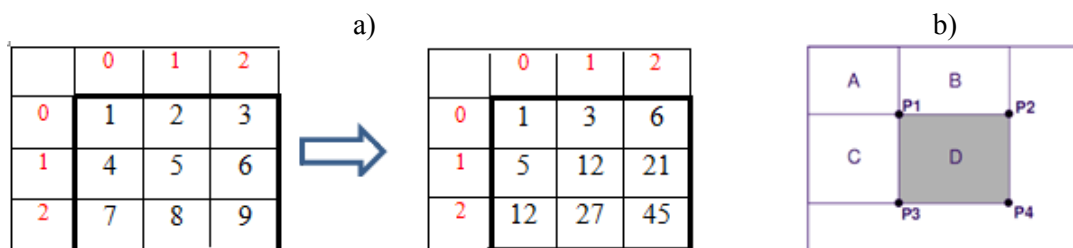
	Wynik pomiaru
$F(x) > 0$	Widoczny obszar, objęty kolorem białym jest jaśniejszy, od obszaru oznaczonego kolorem czarnym.
$F(x) < 0$	Widoczny obszar, objęty kolorem białym jest ciemniejszy, od obszaru oznaczonego kolorem czarnym.
$F(x) = 0$	Widoczny obszar, objęty kolorem białym ma taką samą jasność, jak obszar oznaczony kolorem czarnym.

2.4. Obrazy zsumowane

W celu przyspieszenia obliczeń poszczególnych cech Haara wykorzystano pomocniczą reprezentację obrazu w postaci tzw. obrazów zsumowanych (ang. *Integral image*). „Obraz zsumowany” tworzony jest na podstawie obrazu oryginalnego, przekształconego do skali szarości. Każdy piksel „obrazu oryginalnego” opisany jest za pomocą pojedynczej liczby z zakresu od 0 do 255 (1 bajt). Traktując otrzymany w ten sposób obraz jako macierz prostokątną, jego obraz zsumowany otrzymuje się, wypełniając kolejne komórki tworzonej macierzy w taki sposób, że wartość każdej z nich to suma wartości pikseli obrazu oryginalnego „na lewo i w górę” od danej komórki. Opisane zależności wyraża poniższy wzór [6]:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'), \quad (2)$$

gdzie $i(x', y')$ oznacza wartość pojedynczego piksela obrazu oryginalnego.



Rys. 2. (a) Obrazy zsumowane: a) sposób tworzenia obrazu zsumowanego na podstawie obrazu oryginalnego, zapisanego w skali szarości, b) sposób obliczania sumy wartości jasności pikseli w danym obszarze, z wykorzystaniem obrazu zsumowanego

Fig. 2. Integral images: a) the way of creating the integral image based on the original image, saved in grayscale, b) the way of calculating the sum of pixel brightness values in given area using the integral images

Na rysunku 2a przedstawiono numeryczną postać obrazu oryginalnego po przekształceniu do skali szarości oraz otrzymany na jego podstawie obraz zsumowany.

Obraz zsumowany posiada ciekawe własności pozwalające na znaczne przyspieszenie obliczeń [8]. Przykładowo, chcąc obliczyć sumę wszystkich pikseli z macierzy obrazu oryginalnego, przedstawionej na rysunku 2a, wystarczy tylko jedno odwołanie do komórki o współrzędnych (2,2) macierzy obrazu zsumowanego. Otrzymana wartość to 45.

Podobnie, chcąc obliczyć wartość średniej jasności pikseli w całym obszarze, wystarczy podzielić otrzymaną wartość przez liczbę pikseli w całej macierzy: $45 / 9 = 5$.

Dysponując obrazem oryginalnym, zapisanym w postaci obrazu zsumowanego, można również w bardzo szybki sposób obliczać sumy wartości jasności pikseli w danym fragmencie obrazu.

Rysunek 2b przedstawia obraz zsumowany podzielony na cztery sąsiadujące obszary: A, B, C, D. Punkty: P1, P2, P3, P4 zawierają sumy wartości jasności pikseli w poszczególnych obszarach, co wyrażają poniższe równania [6]:

$$ii(P1) = A \quad (3)$$

$$ii(P2) = A + B \quad (4)$$

$$ii(P3) = A + C \quad (5)$$

$$ii(P4) = A + B + C + D. \quad (6)$$

Z równań: (3), (4), (5), (6), otrzymujemy równanie (7), opisujące sposób obliczenia sumy wartości jasności pikseli w obszarze D, który wymaga jedynie czterech odwołań do macierzy zawierającej wartości obrazu zsumowanego:

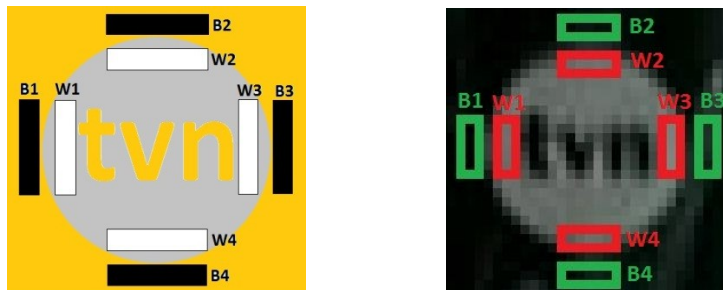
$$D = P4 - P3 - P2 + P1 = P4 - (P3 + P2) + P1. \quad (7)$$

Wykonanie tych samych obliczeń bez wykorzystania obrazu zsumowanego, na prostokątnym obszarze o wymiarach $n \times m$, wymagałoby odczytania i zsumowania $n \times m$ wartości pikseli, co z pewnością trwałoby dłużej niż odczytanie wartości zaledwie w czterech punktach.

2.5. Wykrywanie logo stacji TVN® na podstawie badania różnicy jasności pikseli

Projektując cechę Haara, przeznaczoną do wykrywania logo konkretnej stacji telewizyjnej, należy wcześniej określić jego charakterystyczne cechy, takie jak: kształt, przezroczystość, nasycenie kolorów oraz położenie na ekranie w trakcie emisji. Wnikliwa analiza konkretnego emblematu pozwoli na zaprojektowanie skutecznego filtru, a tym samym zmaksymalizowanie współczynnika wykrywania niechcianych treści. Dla potrzeb niniejszego artykułu do analizy wybrano logo stacji TVN®.

Rysunek 3 przedstawia szkic logo stacji TVN® z nałożoną cechą Haara, opartą na badaniu różnicy jasności w ośmiu wydzielonych obszarach, na granicy przedziałów logo i pozostałej części obrazu.



Rys. 3. Cecha Haara przeznaczona do wykrywania logo stacji TVN® na podstawie różnicy jasności, nałożona na logo stacji. Prostokątami zaznaczono obszary podlegające badaniu. Po lewej stronie widoczny jest rysunek poglądowy, po prawej rzeczywiste logo wycięte z klatki filmu.

Fig. 3. Haar-like feature designated for detecting of TV station logo based on brightness difference, put on station logo. Examined areas have been marked with rectangular areas. From the left: the logo draft, and the real logo, extracted from the frame of a movie.

Egzaminowana klatka materiału video przekształcana jest do skali szarości, a w każdym z prostokątnych obszarów (W1, ..., W4; B1, ..., B4) obliczana jest suma wartości jasności pikseli przy wykorzystaniu obrazów zsumowanych.

Podstawowym założeniem, na którym opiera się prezentowana wyżej cecha Haara, jest obserwacja, że logo stacji TVN® w przeważającej większości przypadków jest jaśniejsze niż obraz znajdujący się w najbliższym sąsiedztwie logotypu. Zasada działania omawianej cechy opiera się na wykrywaniu nagłej zmiany jasności pikseli na krańcach przedziałów logo względem pozostałej części obrazu. Zaobserwowanie takich gwałtownych zmian aż w czterech egzaminowanych i oddalonych od siebie obszarach stwarza bardzo wysokie prawdopodobieństwo obecności logo stacji TVN® na danej klatce badanego materiału.

Na rysunku 3 białymi prostokątami: W1, W2, W3, W4 zaznaczono obszary, w których spodziewana jest większa jasność pikseli. Prostokątami czarnymi: B1, B2, B3, B4 zaznaczono obszary, w których spodziewana jest relatywnie mniejsza jasność pikseli.

Wartość cechy wyraża wzór:

$$F(x) = B1 + B2 + B3 + B4 + W1 + W2 + W3 + W4 - 2(B1 + B2 + B3 + B4) \quad (8)$$

$$P(B1) = P(B2) = P(B3) = P(B4)$$

$$P(W1) = P(W2) = P(W3) = P(W4), \quad (9)$$

gdzie P oznacza pole powierzchni obszaru.

Jak widać ze wzoru (8), współczynnik korekcyjny wynosi 2. Ze względu na fakt, że opisane wzorem (8) prostokąty nie tworzą spójnego obszaru (nie sąsiadują ze sobą), do obliczeń przyjęto wzór (10) w postaci uproszczonej i szybszej obliczeniowo:

$$F(x) = W1 + W2 + W3 + W4 - (B1 + B2 + B3 + B4). \quad (10)$$

Wartość cechy informuje o zależności jasności badanych obszarów względem siebie, zgodnie z tabelą 1. Zakłada się, że logo jest obecne w badanym obszarze tylko wówczas, gdy wartość zmierzonej w tym miejscu cechy $F(x)$ jest większa niż zero.

Niejednoznaczność pomiaru może wystąpić jedynie w sytuacji, gdy logo jest niewidoczne również dla człowieka, to znaczy znajduje się na bardzo jasnym tle (np. biała chmura na błękitnym niebie) lub nad bardzo intensywnym źródłem światła (np. rozbłysk flesza aparatu fotograficznego). Rysunek 4 przedstawia przykładową klatkę, która nie zostanie prawidłowo rozpoznana za pomocą omawianej cechy, pomimo faktycznej obecności logo. Warto zwrócić uwagę na prawy górny róg obrazu – logo nie jest widoczne, pomimo jego rzeczywistej obecności, z powodu bardzo jasnego tła.



Rys. 4. Klatka z niewykrytym logo
Fig. 4. A frame with non detected logo

2.6. Budowa algorytmu klasyfikacji przypadków na podstawie cech Haara

2.6.1. Algorytm klasyfikujący jako *Deterministyczny Automat Skończony*

Na podstawie wartości cechy Haara, opisanej w punkcie 2.3, tworzona jest funkcja klasyfikująca $C(x)$, która może przyjmować wyłącznie wartości bitowe (0 albo 1). Funkcja ta określa, czy w danym momencie trwa blok programowy ($C(x) = 1$) czy też nie trwa blok reklamowy ($C(x) = 0$).

Prezentowany w tym rozdziale algorytm klasyfikuje fragment materiału video na podstawie otrzymanych wartości.

W związku z częściową niejednoznacznością używanej cechy Haara, zdarzają się przypadki błędnej klasyfikacji, rozumiane przez zakwalifikowanie fragmentu bloku programowego jako bloku reklamowego lub na odwrót. Przypadki takie są nazywane fałszywymi trafie-

niami (ang. *false positives*). Z tego powodu algorytm klasyfikujący materiał video musi uwzględniać takie przypadki, inteligentnie rozróżniając fałszywe trafienia od wartości poprawnie sklasyfikowanych.

Zaprojektowany klasyfikator został stworzony na podstawie Deterministycznego Automatu Skończonego DAS (ang. *Deterministic Final Automate*) i jest rozwiązaniem prototypowym, które docelowo zostanie zastąpione algorytmem uczącym.

W zaprojektowanym algorytmie klasyfikującym można wyróżnić cztery stany. Każdy ze stanów posiada swoje oznaczenie numeryczne od 0 do 4, jak następuje:

- 0 – algorytm przebywa w stanie nieokreślonym;
- 1 – algorytm przebywa w stanie oznaczającym blok reklamowy;
- 2 – algorytm przebywa w stanie oznaczającym blok programowy;
- 3 – algorytm przebywa w stanie badania ciągłości przedziału bloku programowego wprzód.

Stan „0” jest stanem początkowym. Algorytm znajduje się w tym stanie tylko na początku swojego działania. „1” i „2” to stany akceptujące. Alfabet wejściowy stanowią wyłącznie wartości bitowe $\{0, 1\}$. Funkcje przejść zdefiniowane są za pomocą wielu warunków.

2.6.2. Opis algorytmu

Do poprawnej pracy algorytm klasyfikujący wymaga określenia czterech parametrów:

1. Minimalnej dopuszczalnej długości przedziału następujących po sobie wartości „1”, oznaczanej przez symbol c , wyrażonej przez ilość sekund, gdzie $c > 0$;
2. Maksymalnej dopuszczalnej ilości występujących po sobie wartości „0” w stanie nr 3, oznaczanej symbolem h , wyrażonej w ilości klatek, gdzie $h > 0$;
3. Minimalnej dopuszczalnej długości bloku reklamowego, oznaczanej przez symbol b , wyrażonej w sekundach, gdzie $b > 0$;
4. Wartości liczbowej oznaczanej symbolem t , wyrażającej wartość cechy Haara, od jakiej dana klatka klasyfikowana jest jako reklama;

Opis algorytmu dzieli się na trzy fazy. W opisie pominięto fazę wstępną, polegającą na zamianie wczytanego zbioru danych, reprezentujących wartość cechy Haara dla kolejnych klatek, na wartości bitowe, zgodnie z parametrem t .

Faza 1

Algorytm rozpoczyna się w stanie oznaczonym symbolem „0”, gdyż w fazie początkowej nie jest możliwe określenie, czy analizowany będzie blok programowy czy też reklamowy. Jeśli na wejście zostanie podana wartość „0”, algorytm przełącza się do stanu reklamowego, w którym pozostaje do czasu otrzymania wartości różnej od „0”. W przypadku otrzymania wartości „1”, algorytm zapamiętuje numer aktualnie przetwarzanej klatki ($f1$) i przełącza się

do stanu nr 3, który ma za zadanie sprawdzenie ciągłości przedziału bloku programowego wprzód. W stanie nr 3 algorytm sprawdza, czy w dalszej części podawanego strumienia bitowego znajduje się ustalona za pomocą parametru c liczba następujących po sobie wartości „1” (warunek $w1$). Dopuszczalne są jedynie przerwania ciągłości (pojawienie się następujących po sobie wartości „0”) o wielkości ustalonej przez parametr h (warunek $w2$). Ma to na celu ominięcie fałszywych trafień, które wprowadzają pewne zaburzenie analizowanych danych. Algorytm pozostaje w stanie nr 3 dopóki spełnione są obydwie z wymienionych warunków ($w1$ i $w2$). W razie ich pomyślnego sprawdzenia (analiza warunków wymaga przebywania w stanie nr 3 przez pewien czas), algorytm przełącza się do stanu nr 2 (blok programowy), oznaczając jednocześnie wcześniej zapamiętaną wartość ($f1$) jako początek potencjalnego bloku programowego. Jeśli, będąc w stanie nr 3, algorytm stwierdzi, że analizowany ciąg wartości „1” nie spełnia któregoś z opisanych warunków ($w1$ lub $w2$), następuje powrót do stanu nr 1 (blok reklamowy), a zapamiętany numer klatki kandydata jest kasowany.

Algorytm pozostaje w stanie nr 2 dopóty, dopóki na wejście podawana jest wartość „1” (blok reklamowy). W momencie otrzymania wartości „0” algorytm przełącza się w stan nr 1, zapisując równocześnie numer klatki kończącej dany blok programowy ($f2$).

W wyniku działania opisanej procedury otrzymywany jest wektor struktur ($v1$), z których każda zawiera klatkę początkową i klatkę końcową ciągłego bloku danych o dopuszczalnym przerwaniu ciągłości nie większym niż określa parametr h . W razie potrzeby, do tego wektora dołączane są osobno bloki krańcowe (występujące na początku i końcu badanego materiału video).

Bloki przechowywane w wektorze ($v1$) stanowią kandydatów na bloki programowe.

Faza 2

W dalszym etapie następuje próba filtracji bloków reklamowych spośród otrzymanych bloków programowych. Procedura ta polega na badaniu odstępów pomiędzy blokami, wyrażonych ilością klatek. Jeśli liczba klatek pomiędzy blokami programowymi jest większa niż określa to parametr b , to blok danych znajdujący się pomiędzy nimi jest klasyfikowany jako potencjalny blok reklamowy i zapisywany w osobnym wektorze.

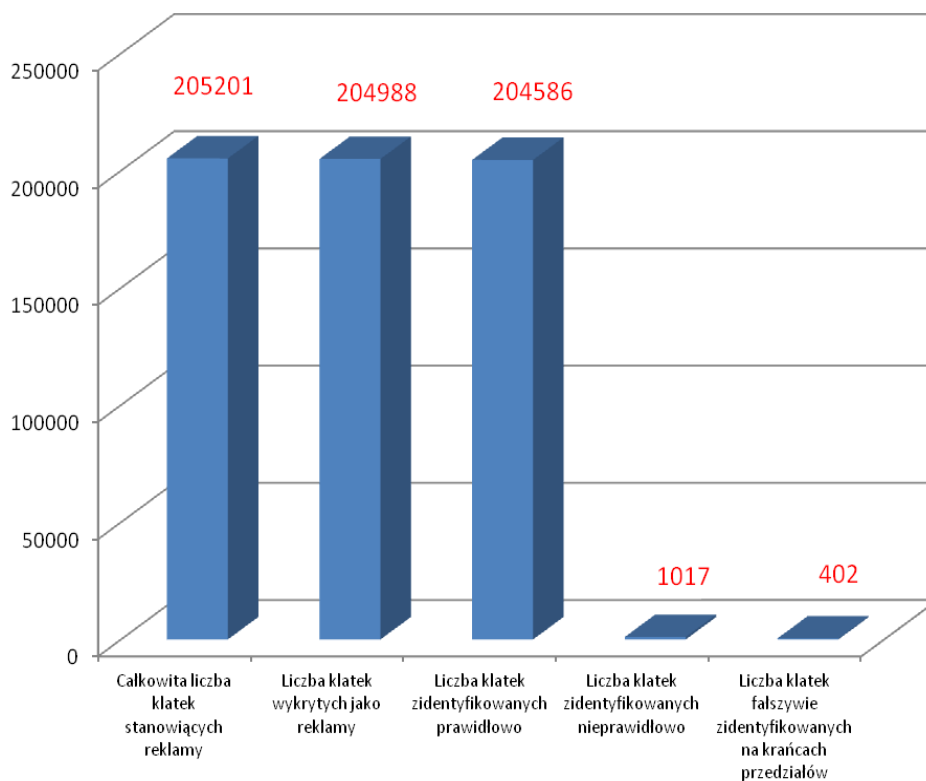
Na końcu tej procedury otrzymywany jest kolejny wektor danych, zawierający potencjalne bloki reklamowe ($v2$).

Faza 3

Otrzymane bloki poddawane są procedurze badania procentowej ilości wartości „0” i „1”. Jeśli algorytm stwierdzi, że w dowolnym z potencjalnych bloków reklamowych/ ilość wartości „1” jest większa od liczby wartości „0”, blok taki klasyfikowany jest jako fałszywe trafienie i kasowany z wektora ($v2$).

Ostatecznie, pozostałe bloki wektora (v2) klasyfikowane są jako odnalezione bloki reklamowe. Algorytm kończy działanie.

3. Wyniki testów



Rys. 5. Statystyki wykrywalności reklam

Fig. 5. Statistics of TV commercials detectability

Tabela 2

Wyniki testów wydajnościowych procesu wykrywania reklam

Nr	Czas wykrywania reklam	Czas trwania materiału video	Liczba klatek analizowanych w ciągu jednej sekundy	Stosunek czasu trwania materiału video do czasu wykrywania reklam
1	00:05:14	01:59:55	572.14	22,91
2	00:01:29	00:34:56	590.47	23,55
3	00:02:39	00:59:45	564.86	22,54
4	00:04:59	01:59:05	596.54	23,90
5	00:02:25	00:59:57	619.81	24,80
6	00:02:29	00:59:57	604.92	24,14
7	00:02:24	00:59:57	622.96	24,98
8	00:04:37	01:48:37	588.68	23,53
9	00:05:04	02:05:33	618.74	24,78

W ramach niniejszych badań zaprojektowano i zrealizowano (przy użyciu języka C++ oraz biblioteki Intel® Integrated Performance Primitives [10] w wersji 6.1.6.056) prototypową aplikację, w której zaimplementowano opisane wyżej algorytmy.

Przeprowadzono wiele testów mających na celu praktyczne sprawdzenie poprawności i szybkości działania opracowanych metod detekcji reklam telewizyjnych. Procedura testowa opierała się na oznaczeniu stosunku ilości materiału video zakwalifikowanego przez program jako treści reklamowe do ilości realnych treści reklamowych, występujących w danym pliku. Wyniki przedstawione zostały na rysunku 5.

Przeprowadzono również pomiary liczby klatek video analizowanych przez aplikację w ciągu jednej sekundy. Wyniki zostały zamieszczone w tabeli 2.

4. Podsumowanie

Z przeprowadzonej serii testów wynika, że zaprojektowane metody wykrywania treści reklamowych działają bardzo sprawnie, a stworzona aplikacja, dzięki dostosowaniu do cech konkretnej stacji telewizyjnej, osiąga wysoki poziom skuteczności (rzędu 99,70%). W przeważającej większości wypadków program odnalazł wszystkie obecne w nagrany materiale reklamy telewizyjne i oznaczył prawidłowo ich granice, z dokładnością do jednej klatki. Fakt ten dowodzi efektywności zaprojektowanego algorytmu klasyfikującego, a także jego odporności na tzw. fałszywe trafienia. Nieprawidłowo sklasyfikowane przypadki można jednak łatwo skorygować w przyszłości, lepiej dobierając zestaw proponowanych cech.

Na uwagę zasługuje również fakt, że zaprojektowany algorytm do analizy video jest wydajny i działa szybko, biorąc pod uwagę bardzo wysoką dokładność procesu detekcji. Fakt ten umożliwia wykrywanie reklam telewizyjnych w czasie rzeczywistym, np. w trakcie oglądania materiału video.

BIBLIOGRAFIA

1. Rowe L. A., Boreczky J. S., Eads C. A.: Indexes for User Access to Large Video Databases. IS & / SPIE Symposium on Electronic Imaging Science and Technology, Conference, 1994.
2. Aas K., Eikvil L.: Indexes for User Access to Large Video Databases. 1997.
3. Brunelli R., Mich O., Modena C. M.: A Survey on the Automatic Indexing of Video Data. Journal of Visual Communication and Image Representation 10, Trento, Italy 1999.

4. Snoek C. G. M., Worring M.: Concept-Based Video Retrieval. *Journal Foundations and Trends in Information Retrieval*, Vol. 2, Issue 4, April 2008.
5. Thi-Lan L., Boucher A., Thonnat M., Bremond F.: A framework for surveillance video indexing and retrieval. *Content-Based Multimedia Indexing*, 2008.
6. Viola P., Jones M. J.: Rapid Object Detection using a Boosted Cascade of Simple Features. *IEEE CVPR*, 2001.
7. Papageorgiou C. P., Oren M., Poggio T.: A General Framework for Object Detection. *International Conference on Computer Vision*, 1998.
8. Gąciarz T.: Fast Adaptive Binarisation for Camera-based Document Images. *Proceedings of International Conference on Artificial Intelligence, IIS Kraków 2009*.
9. Gąciarz T., Czajkowski K.: *Klasyfikacja zdjęć w multimedialnych bazach danych. Studia Informatica*, Vol. 31, No. 2A (89), Wydawnictwo Politechniki Śląskiej, Gliwice 2010.
10. Nayak A., Stojmenović I.: *Hanbook of Applied Algorithms – Solving Scientific, Engineering and Practical Problems*. Wiley-Interscience, New Jersey 2008.
11. Schapire R. E.: *The Boosting Approach To Machine Learning, An Overview*. *Lecture Notes In Statistics*, Springer Verlag, New York 2002.
12. Freund Y., Schapire R. E.: A decision-theoretic generalization of on-line learning and an application to boosting. *Computational Learning Theory: Eurocolt 95*, Springer-Verlag, 1995, s. 23÷37.
13. Wang J., Duan L., Li Z., Liu J., Lu H., Jin J. S.: A robust method for TV logo tracking in video streams. *Multimedia and Expo, IEEE International Conference*, 2006.
14. Marlow S., Sadlier D. A., McGeough K., O'Connor N., Murphy N.: *Audio and Video Processing for Automatic TV Advertisement Detection*. *ISSC 2001 – Irish Signals and Systems Conference*, 25-27 June 2001, Maynooth, Ireland 2001.
15. Dimitrova N., Jeannin S., Nesvadba J., McGee T., Agnihotri L., Mekenkamp G.: *Real time commercial detection using MPEG features*. *Visual Information Processing (EUVIP)*, 2010.
16. Li Y., Zhang D., Zhou X., Jin J. S.: *A Confidence Based Recognition System for TV Commercial Extraction*. *Proc. Nineteenth Australasian Database Conference (ADC 2008)*, Wollongong, NSW, Australia 2008.
17. Stewart T.: *Intel® Integrated Performance Primitives – How to Optimize Software Applications Using Intel® IPP*. Intel Press, 2004.
18. Crow F.: *Summed-area tables for texture mapping*. *Proceedings of SIGGRAPH*, Vol. 18(3), 1984, s. 207÷212.

19. Lienhart R., Maydt J.: An extended set of Haar-like features for rapid object detection. Proceedings of the International Conference on Image Processing, 2002.
20. Wang J., Duan L. et al.: A robust method for TV logo tracking in video streams. IEEE International Conference on Multimedia and Expo, 2006.
21. Marlow S., Sadlier D. A., McGeough K., O'Connor N., Murphy N.: Audio and Video Processing for Automatic TV Advertisement Detection, ISSC 2001.
22. Dimitrova N., Jeannin S., et al.: Real time commercial detection using MPEG features. International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Annecy, France 2002, s. 1÷6.
23. Li Y., Zhang D., Zhou X., Jin J. S.: A Confidence Based Recognition System for TV Commercial Extraction. ADC, Vol. 75, Australian Computer Society, 2008.

Wpłynęło do Redakcji 8 stycznia 2012 r.

Abstract

Video databases generate serious challenges in obtaining, storing and processing data. One of the biggest problem is searching for information in such databases. This situation causes many problems while trying to save in the database movies emitted by TV. TV commercials, which are present in recorded material, unnecessarily enlarge the size of a generated file and irritate recipient by necessity of manual searching of a record for finding desirable content during show. It is worth to notify, that manual extracting of TV commercials from video files (despite theoretical possibility), takes a long time and requires the use of specialized editing tools. These tools, although effective, usually are not free and their price is not commensurate to functionality expected in discussed case. The solution of presented problems is creation of algorithms, which are able to automate processes, associated with getting rid of unwanted TV commercials.

A broadcaster TV logo is the most stable property, which allows to distinguish regular content from commercials content. So called Haar-like features have been used to TV logo detection process. Haar-like features are kind of programmable graphic masks, which are put on given graphic image to check if a requested object is present in the appointed area.

For purposes of this paper, the prototype computer application, has been designed. It uses described above algorithms. The purpose of the application was to determine the presence or absence of the TV logo station on examined frame of video material. In next step, a few tests

to check in practice correctness and quickness of work of showed TV commercials detection methods have been executed.

The test results proves that designed method of TV commercials detection works very accurately. Created application achieves high level of precision (about 99,70%), due to ability to adjust the parameters to the particular TV station properties. It is very good result, comparing to other currently available solutions. The presented solutions are good starting point to develop efficient and effective application framework, designed to detect and track predefined objects on graphic images as well in video databases.

Adresy

Michał MAJEWICZ: Politechnika Krakowska, Wydział Fizyki, Matematyki i Informatyki, Instytut Teleinformatyki, ul. Warszawska 24, 31-155 Kraków, Polska, majewicz@mars.iti.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.

Krzysztof CZAJKOWSKI: Politechnika Krakowska, Wydział Fizyki, Matematyki i Informatyki, Instytut Teleinformatyki, ul. Warszawska 24, 31-155 Kraków, Polska, kc@pk.edu.pl.