

XIII Seminarium
ZASTOSOWANIE KOMPUTERÓW W NAUCE I TECHNICIE 2003
Oddział Gdański PTETiS

SYSTEM SYNCHRONIZACJI NAGRAŃ WIELOKANAŁOWYCH

Przemysław MAZIEWSKI, Piotr SZCZUKO

Katedra Systemów Multimedialnych, Wydział Elektroniki, Telekomunikacji i Informatyki,
Politechnika Gdańska, ul. G. Narutowicza 11/12, 80-952 Gdańsk
tel: (058) 347-23-01 e-mail: przemas@sound.eti.pg.gda.pl,
szczuko@sound.eti.pg.gda.pl

Niniejsza praca zawiera opis komputerowego systemu synchronizacji wielu dźwięków pochodzących z jednego źródła i nagranych w kilku różnych, niesynchronizowanych urządzeniach rejestrujących. System wykorzystuje korelację skrośną.

1. WPROWADZENIE

Praca opisuje synchronizację plików dźwiękowych, uzyskanych poprzez równoczesną rejestrację sygnału w wielu odbiornikach położonych w różnych odległościach od źródła, przy braku synchronizacji pomiędzy nimi i w odmiennych warunkach szumowych. Podczas odtwarzania dwóch zarejestrowanych plików w stereofonicznym systemie odsłuchowym, nastąpi zaburzenie prawidłowego postrzegania położenia źródła dźwięku (zaburzenia w procesie lokalizacji dźwięku opisane krzywą ITD – ang. *Interaural Time Differences*). W celu dokładnego umieszczenia pozornego źródła dźwięku w pozycji centralnej konieczne jest wykonanie synchronizacji obu nagrań.

2. PODSTAWY TEORETYCZNE METODY KORELACJI SKROŚNIEJ

Ciąg korelacji pomiędzy dwoma sygnałami rzeczywistymi $x_1[n]$ i $x_2[n]$ jest definiowany jako zależność dana wzorem (1):

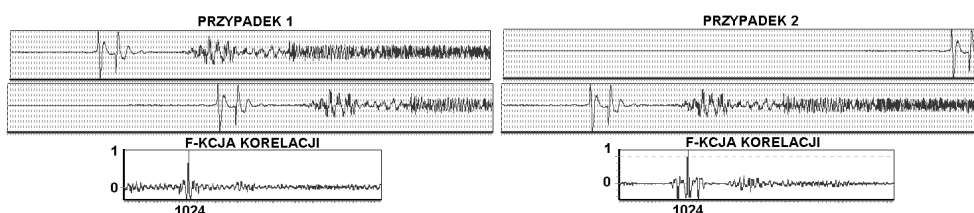
$$R_{x_1x_2}[m] = E\{x_1[n]x_2[n-m]\} \quad (1)$$

gdzie $E\{ \}$ oznacza wartość oczekiwaną. Podana zależność może być estymowana dla skończonych obserwacji sygnałów $\{x_1[n]\}$ i $\{x_2[n]\}$ za pomocą następującej zależności:

$$\hat{R}_{x_1 x_2}[m] = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} x_1[n] x_2[n-m] \quad (2)$$

Położenie maksimum korelacji wyrażone jako funkcja opóźnienia o m próbek, może być interpretowane jako estymata opóźnienia czasowego (TDE – ang. *Time Delay Estimation*) pomiędzy badanymi sygnałami. Określone opóźnienie jest zawsze wyrażone w całkowitej liczbie próbek. W literaturze istnieją publikacje opisujące możliwość uzyskania dokładności estymaty na poziomie sub-próbek (ułamku okresu próbkowania) [1]. Rozwiązania takie bazują na odpowiednio przeprowadzonej interpolacji funkcji korelacji skrośnej (np. za pomocą funkcji sinc lub funkcji polinomialnych) [4] [5].

W przypadku użycia korelacji do określenia opóźnienia pomiędzy dwoma ciągami próbek dźwiękowymi występuje następujący problem. Położenie maksimum funkcji nie niesie pełnej informacji o tym, który z dźwięków odtwarzanych z plików poprzedza inne na osi czasu. Taki przypadek zilustrowany jest na rysunku 1.



Rys. 1 Dwa przypadki prowadzące do identycznego położenia maksimum w otrzymanych ciągach korelacji

Aby określić względne przesunięcie pomiędzy dwoma plikami zastosowano następujące rozwiązanie. Każdy z plików wejściowych został podzielony na dwie równe części. Następnie zbadano stopień korelacji (maksymalną wartość ciągu korelacji) pomiędzy poszczególnymi częściami pierwszego i drugiego pliku. Otrzymane dane zapisano w postaci macierzy kwadratowej (2 na 2). W zależności od położenia maksimum funkcji korelacji skrośnej i maksimum w otrzymanej macierzy można precyzyjnie określić względne przesunięcie pomiędzy dwoma badanymi plikami dźwiękowymi.

Istnieje kilka algorytmów pozwalających na praktyczne wyliczenie funkcji korelacji skrośnej. Na potrzeby niniejszego rozwiązania użyto algorytmu bazującego na FFT [2] [3].

Jeśli $\{x_1[n]\}$ i $\{x_2[n]\}$ oznaczają skończone realizacje sygnałów dla $n=0..N-1$, ich widma można wyliczyć za pomocą szybkiej transformaty Fouriera.

$$\{S_1[k]\} = FFT\{\{x_1[n]\}\} \quad (3)$$

$$\{S_2[k]\} = FFT\{\{x_2[n]\}\} \quad (4)$$

gdzie: $k=0..N-1$, N - potęga 2-ki

Na podstawie widm S_1 i S_2 wylicza się „widmo skrośne” dane poniższą zależnością:

$$\{S_{12}[k]\} = \{S_1[k]\} \cdot \{S_2[k]\} \quad (5)$$

gdzie: $k=0..N-1$, N - potęga 2-ki

Po wyznaczeniu widma S_{12} liczona jest estymata funkcji korelacji skrośnej, jako odwrotna transformata Fouriera.

$$\{R_{x_1x_2}[r]\} = IFFT(\{S_{12}[k]\}) \quad (6)$$

3. PRAKTYCZNA REALIZACJA ALGORYTMU BAZUJĄCEGO NA KORELACJI SKROŚNEJ

Praktyczna realizacja przedstawionego algorytmu została dokonana w środowisku Matlab. Poszczególne kroki od momentu wczytania plików PCM do uzyskania wartości przesunięcia czasowego przedstawiają się następująco:

- z wczytanych dwóch plików $\{x_1[n]\}$, $n=0..N$ oraz $\{x_2[m]\}$, $m=0..M$, eliminowana jest składowa stała:

$$\{x_3[n]\} = \{x_1[n]\} - \bar{x}_1[n]; \{x_4[m]\} = \{x_2[m]\} - \bar{x}_2[m] \quad (7)$$

- wyliczenie dyskretnej transformaty Fouriera (DFT) obu przebiegów :

$$\begin{aligned} \{X[k]\} &= DTF(\{x_3[n]\}) \\ \{Y[k]\} &= DTF(\{x_4[m]\}) \end{aligned} \quad (8)$$

- wyliczenie ciągów autokorelacji:

$$\begin{aligned} \{A_x\} &= IDTF(\{X\} \cdot \{X\}^*) \\ \{A_y\} &= IDTF(\{Y\} \cdot \{Y\}^*) \end{aligned} \quad (9)$$

- wyliczenie zespolonego ciągu korelacji :

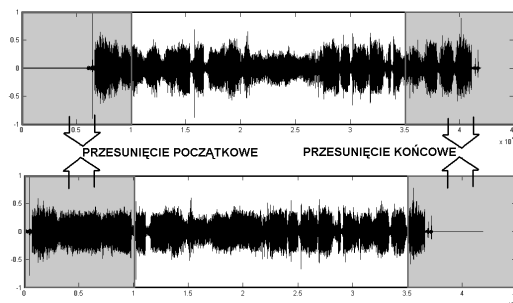
$$\{A_{XY}\} = IDTF(\{X\} \cdot \{Y\}^*) \quad (10)$$

- wyliczenie ciągu korelacji :

$$\{R_{x_1x_2}\} = \frac{\text{real}(A_{XY})}{\sqrt{A_x[0]} \cdot \sqrt{A_y[0]}} \quad (11)$$

Korzystając z opisanego powyżej algorytmu estymatora ciągu korelacji stworzono pierwszy „prosty” algorytm synchronizacji. Działanie tego algorytmu zobrazowano na rysunku 2.

Korelacja liczona jest pomiędzy początkowymi i końcowymi buforami obu plików (zaznaczonymi na szaro na rysunku 2). Na podstawie otrzymanych wartości przesunięć pomiędzy początkiem i końcem plików wyeliminowano: początkowe przesunięcie (poprzez odpowiednie przesunięcie drugiego z plików na osi czasu), różnice w długości obu plików (poprzez przepróbkowanie sygnału zapisanego w pliku).

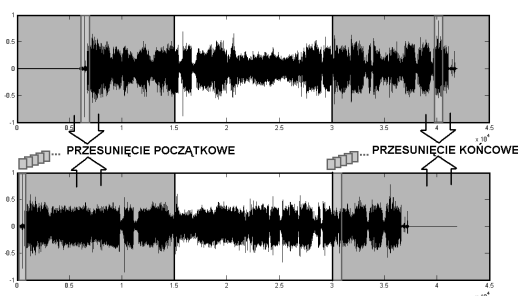


Rys. 2 Ilustracja działania prostego algorytmu synchronizacji

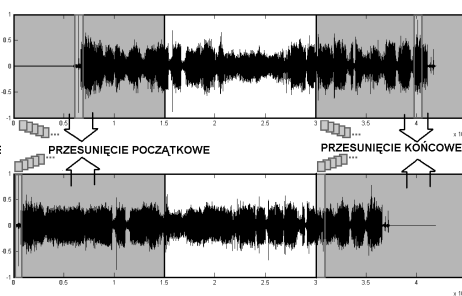
Pierwsza modyfikacja „prostego algorytmu” była następująca: w buforze początkowym pliku referencyjnego wyznaczano położenie pewnej cechy dystynktywnej. Ta cecha dystynktywna rozumiana jest jako krótki fragment pliku dźwiękowego charakteryzujący się odpowiednią szczytową wartością chwilową i odpowiednią wartością skuteczną. Wyszukiwanie takiego fragmentu polega na wyszukaniu ramki „P”, zawierającej określoną liczbę próbek. Ramka jest wybierana na zasadzie porównywania wartości bezwzględnych i wartości skutecznych w ramach (algorytm pozwala ustalić wagi dla tych dwóch czynników). Ramki te mają określoną długość i wybierane są z bufora wejściowego ze zmiennym krokiem nakładkowania. W kolejnym kroku bada się korelację ramki „P” z kolejnymi ramkami bufora wejściowego drugiego pliku wybieranymi w ten sam sposób, jak w przypadku wyszukiwania ramki „P”. Ramki o największej wartości funkcji korelacji wyznaczały przesunięcie czasowe pomiędzy buforami. W podobny sposób analizowane były bufor końcowe synchronizowanych plików.

Ilustrację działania opisanego algorytmu przedstawia rysunek 3.

Druga modyfikacja polegała na odrzuceniu kroku wyszukiwania cechy dystynktywnej w buforze początkowym. W zamian badano korelację pomiędzy każdą z ramek buforów pierwszego i drugiego pliku. Zasada działania trzeciego algorytmu prezentowana jest na poniższym rysunku (rys. 4).



Rys. 3 Ilustracja działania zmodyfikowanego algorytmu synchronizacji (algorytm drugi)



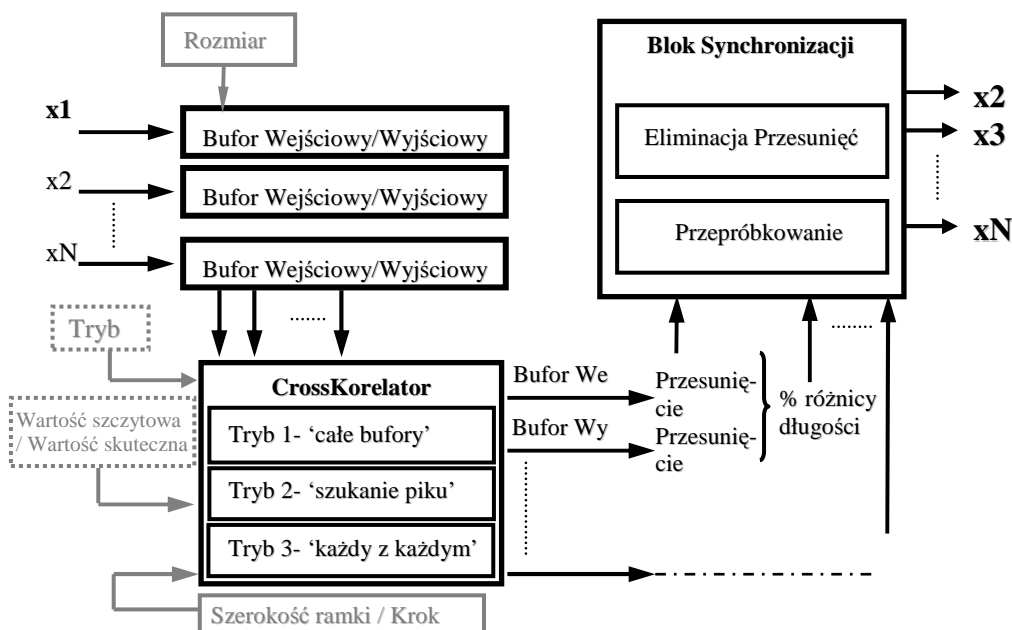
Rys. 4 Ilustracja działania zmodyfikowanego algorytmu synchronizacji (algorytm trzeci)

Podsumowując, działanie trzeciego z opisywanych algorytmów wygląda następująco: bufor (zaznaczone na szaro na rysunkach powyżej) dzielone są na ramki o określonej długości (parametr aplikacji) i kroku nakładkowania (parametr aplikacji wyskalowany w liczbie próbek). W następnym etapie liczona jest korelacja skrośna pomiędzy



każdą z ramek buforów pierwszego i drugiego pliku. Wartość korelacji skróśnej pomiędzy wybranymi przez algorytm ramkami interpretowana jest jako stopień poprawności synchronizacji. Zasada działania opisywanego algorytmu prezentowana jest na schemacie – rysunek 5.

W celu wyrównania długości sygnałów wykonywane jest próbkowanie jednego z nich. Zastosowano metodę liniową: wartości sygnału między próbkami wyznaczone są poprzez interpolację liniową. Ten sposób próbkowania wprowadza niewielki szum do sygnału, który jednak nie jest znaczący w danych zastosowaniach.



Rys. 5 Schemat działania algorytmu synchronizacji

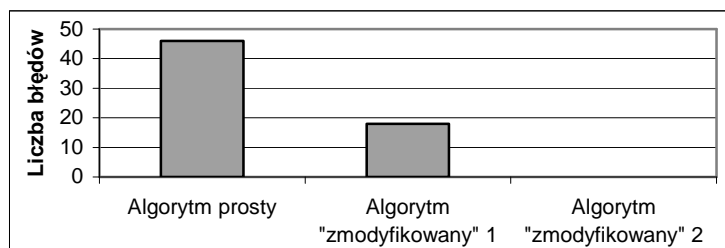
4. BADANIA EKSPERYMENTALNE I ICH WYNIKI

Testom poddano opisane w poprzednim punkcie algorytmy synchronizacji. Jako materiał badawczy posłużyły następujące pliki dźwiękowe:

- plik oryginalny zawierający próbki mowy (pasmo do 8kHz),
- plik oryginalny przesunięty na osi czasu,
- plik oryginalny przesunięty na osi czasu z addytywnym szumem brązowym,
- plik oryginalny przesunięty na osi czasu z addytywnym zmodyfikowanym szumem, białym (modyfikacja polegała na przefiltrowaniu filtrem środkowo-przepustowym 200Hz-8kHz),
- plik oryginalny skrócony o 6% i przesunięty na osi czasu,
- plik oryginalny wydłużony o 6% i przesunięty na osi czasu,
- plik oryginalny z wprowadzonym przesterowaniem, przesunięty na osi czasu,
- plik oryginalny z filtracją dolno-przepustową (2kHz), przesunięty na osi czasu,
- plik oryginalny z filtracją górno-przepustową (300Hz), przesunięty na osi czasu,
- plik oryginalny z dodatkowymi zakłóceniami impulsowymi, przesunięty na osi czasu.



Powyższe pliki wprowadzono parami (oryginał, zmodyfikowany) do poszczególnych algorytmów synchronizacji. Zbadano w ten sposób 70 kombinacji plików wejściowych. Prezentowany na rysunku 6 wykres przedstawia liczbę błędów w synchronizacji dla poszczególnych algorytmów.



Rys. 6 Liczba błędów synchronizacji dla poszczególnych algorytmów

Wszystkie błędy dla „prostego algorytmu” wystąpiły w przypadku plików o zmodyfikowanym czasie trwania (równomierne próbkowanie). Drugi („zmodyfikowany” 1) z algorytmów wykazał błędy w następujących przypadkach: brak cechy dystynktywnej w obu badanych plikach, duże zaszumienie i zniekształcenia impulsowe pliku zmodyfikowanego, filtracja i przesterowanie w pliku zmodyfikowanym. Błędy w tym algorytmie wynikały również z nieodpowiedniego dobrania wag (szczytowa wartość chwilowa, wartość skuteczna) przy wyborze ramki z cechą dystynktywną. Trzeci algorytm („zmodyfikowany” 2) przeszedł pomyślnie wszystkie przeprowadzone próby.

PODZIĘKOWANIA

Opisane eksperymenty badawcze zostały dofinansowane przez Ministerstwo Nauki i Informatyzacji w ramach projektu nr 113/BO/B.

5. BIBLIOGRAFIA

1. Hicks C. M.: Multichannel Modelling of Audio Signals, PhD thesis, Cambridge University Engineering Department, 1999
2. Dolia A. N., Lukin V., Zelensky A.: Fast algorithm of cross-correlation output calculation for interferometric system, Englewood Cliffs, N.J., Prentice-Hall Inc., 1975
3. Rabiner L.R., Gold B.: Theory and Applications of Digital Signal Processing, Englewood Cliffs, N.J., Prentice-Hall Inc., 1975.
4. Hicks Ch.: Resynchronization of multiple Audio Sources, Proc. AES 20th, Budapest, 2001
5. Hicks Ch, Godsill S. J.: A dual-channel approach to the removal of impulse noise from archive recordings, Proc. IEEE ICASSP, 1994

A SYSTEM FOR MULTI-CHANNEL SOUND SYNCHRONIZATION

The aim of this work is to present a computer-based system for multi-channel sound synchronization. Multiple sounds are simultaneously recorded in different recorders and without synchronization between them. Computing is based on cross-correlation techniques. The discussion of the implementation and results of tests are included.

