

Instrukcja

Analiza Sygnału  
w Dziedzinie  
Czasu i Częstości  
dr Teodor Buchner

POLITECHNIKA WARSZAWSKA  
WYDZIAŁ FIZYKI

PRACOWNIA FIZYKI UKŁADU KRĄŻENIA CZŁOWIEKA  
[efizyka.if.pw.edu.pl/twiki/bin/view/ACC/](http://efizyka.if.pw.edu.pl/twiki/bin/view/ACC/)



KAPITAŁ LUDZKI  
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



PRZYGOTOWANIE I REALIZACJA SPECJALNOŚCI WSPÓLFINANSOWANE ZE ŚRODKÓW UNII EUROPEJSKIEJ W RAMACH EUROPEJSKIEGO FUNDUSZU SPOŁECZNEGO

## Ćwiczenie 1 *Jaka to melodia?*

### Wprowadzenie

Instrumenty muzyczne dzielimy na dęte strunowe i perkusyjne. W instrumentach strunowych źródłem dźwięku jest struna szarpana (np gitara) lub uderzana (np. fortepian). Teoria opisująca drgania, które zostają wzbudzone w sztywnej strunie jest bardzo złożona (por. <https://ccrma.stanford.edu/~jos/PianoString/PianoString.html>), natomiast w pierwszym przybliżeniu można opisać te drgania za pomocą liniowego równania falowego.

$$\frac{\partial^2 y}{\partial t^2} = \nu^2 \frac{\partial^2 y}{\partial x^2} \quad (1)$$

(por. także <http://kr.cs.ait.ac.th/~radok/physics/i4.htm>) Prędkość rozchodzenia się fali można wyrazić jako:

$$\nu = \sqrt{\frac{T_0}{\rho_0}} \quad (2)$$

gdzie  $T_0$  - napięcie struny w położeniu równowagi, zaś  $\rho_0$  - gęstość liniowa (masa jednostki długości) struny. Dla danej struny wartości te są stałe, co oznacza że w pierwszym przybliżeniu prędkość dźwięku w strunie można uznać za stałą, niezależną od wysokości tego dźwięku. W konsekwencji w strunie mogą zostać wzbudzone takie drgania, dla których na obu końcach struny będą znajdować się węzły fali, zaś na długości struny będzie się znajdować nieparzysta wielokrotność połówek długości fali. Długość fali jest związana z jej częstotliwością w sposób następujący:

$$\nu = \lambda f \quad (3)$$

Jeśli spoczynkowa długość struny (tzw menzura) wynosi  $L$  (dla swobodnej struny gitary akustycznej menzura wynosi  $L = 620 \text{ mm}$ , dla elektrycznej  $L = 624 \text{ mm}$ ), w takim razie możemy spodziewać się wzbudzenia fal o następujących długościach ( $k$  oznacza ułamek  $\frac{1}{k}$  długości struny odpowiadający połowie długości fali):

$$\lambda_k = \frac{2L}{k} \quad (4)$$

Odpowiada to częstotliwościom:

$$f_k = \frac{\nu}{2L} \cdot k \quad (5)$$

Jeżeli częstość podstawową dla  $k = 1$  oznaczymy jako  $f_1$ , widać że w strunie mogą powstać kolejne wielokrotności tej częstości - tzw *częstotliwości wyższe harmoniczne (overtones)*.

# Elementy akustyki: strój równomierny, temperowany

Podstawowym interwałem stosowanym w akustyce i analizie harmoniczej jest oktawa. Dwa dźwięki różnią się o oktawę jeśli ich częstości mają się do siebie jak 1:2. Jeśli ton podstawowy to  $f_1$  to ton o oktawę niższy będzie miał częstość  $\frac{f_1}{2}$ , zaś ton o oktawę wyższy:  $2f_1$ . Ton o dwie oktawy wyższy to częstość  $4f_1$ , zaś o dwie oktawy niższy  $\frac{f_1}{4}$  i tak dalej. Oktawa dzielona jest na dwanaście półtonów. Nazwy tych półtonów to:

$$C_m, C\#_m, D_m, D\#_m, E_m, F_m, F\#_m, G_m, G\#_m, A_m, A\#_m, B_m,$$

gdzie  $m$ -numer oktawy. Częstości półtonów w stosunku do  $C_m$  w historii bywały dobierane różnie, natomiast od XIX wieku wprowadzono powszechnie tzw strój równomiernie temperowany (equally tempered) oparty na tonie podstawowym  $A_4 = 440\text{Hz}$  (warto zwrócić uwagę że wtedy kolejne nuty oznaczone są literami  $A, B, C, D, E, F, G$ ). System równomiernie temperowany jest to strój muzyczny, który polega na podziale oktawy na dwanaście równych półtonów. Stosunek częstości dwóch kolejnych dźwięków w systemie równomiernie temperowanym wynosi  $\sqrt[12]{2}$  (Wiki). Za ton podstawowy przyjmuje się ton  $A_4$  (dźwięk A w 4-tej oktawie) o częstości  $f_A = 440\text{ Hz}$ . W muzyce istnieje jeszcze wiele różnych strojów (Wiki). Oktawy są numerowane liczbami naturalnymi od 0. Przykładowo zakres dźwięków w pianinie obejmuje oktawy od 0 ( $A_0$ ) do 7 ( $A_7$ ). Częstość każdego dźwięku  $f_p$  można wyrazić w liczbie półtonów  $p$ , które dzielią go od dźwięku  $A_4$ :

$$f_p = f_A \cdot 2^{\frac{p}{12}} \quad (6)$$

$$p = 12 \cdot (\ln_2 f - \ln_2 f_A) \quad (7)$$

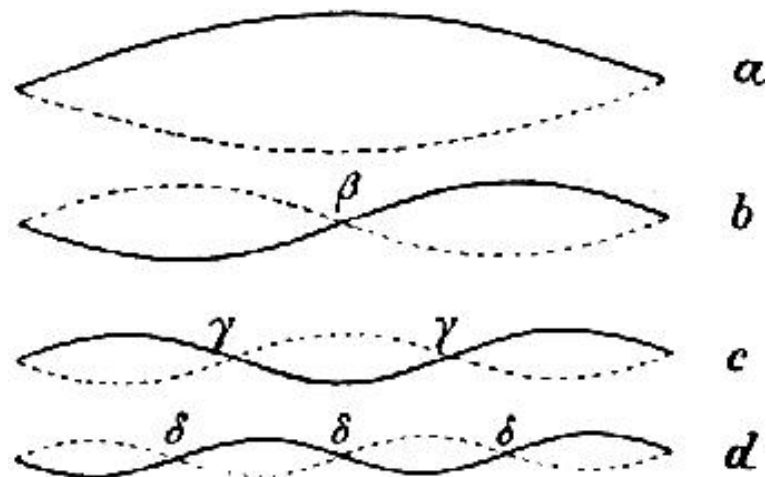
Widać że skala dźwięków którą człowiek uznaje za równomierną jest w istocie skalą logarytmiczną. Dzieje się tak ponieważ ucho ludzkie nie percepuje bezwzględnej różnicy częstości  $\Delta f$  a jedynie względną:  $\frac{\Delta f}{f}$ , w związku z tym odstęp między kolejnymi dźwiękami rosną proporcjonalnie do częstości tych dźwięków. Z relacji (??) można łatwo pokazać iż względna różnica częstości sąsiednich półtonów jest stała, niezależna od ich częstości i równa:  $(2^{\frac{1}{12}} - 1)$ .

Ciekawa jest obserwacja położenia kolejnych harmonicznym tonu podstawowego na skali równomiernie temperowanej. Tabela (??) przedstawia rząd harmoniczej oraz interwał tej harmoniczej od tonu podstawowego, wyrażony w liczbie półtonów, zaś wyliczony wg wzoru (??) jako  $n_k = 12 \ln_2 k$ . Dodatkowo podana jest nazwa interwału używana w muzyce. Tam gdzie interwał ten jest przybliżony, nazwa podana jest w nawiasie.

Rząd harmoniczej $k$	Interwał $n_k$ w półtonach	Stosunek częstości	Nazwa interwału
1	0	1:1	pryma
2	12	2:1	oktawa
3	19.020	$\sim 3 : 2$	(oktawa + kwinta)
4	24	4:1	2 oktawy
5	27.863	$\sim 7 : 3$	(2 oktawy + tercja)
6	31.020	$\sim 5 : 2$	(2 oktawy + kwinta)
7	33.688	$\sim 14 : 5$	(2 oktawy + septyma)
8	36	8:1	3 oktawy

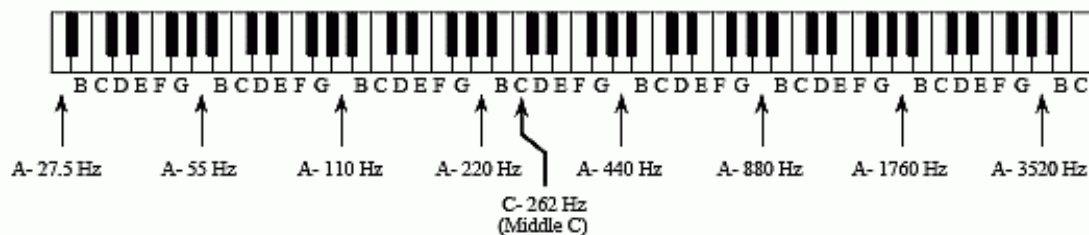
Tablica 1: Harmoniczne tonu podstawowego na skali równomiernie temperowanej

Jak widać z tabeli (??), wyższe harmoniczne częstości podstawowej przypadają na dźwięki oddalone od tonu podstawowego o jedną lub kilka oktaw ( $k = 1, 2, 4, 8$ ) lub też w przybliżeniu o oktawę lub kilka oktaw i kwintę ( $k = 3, 6, 12$ ). Dzięki temu że odstojenie częstości jest żadne w przypadku oktaw a niewielkie w przypadku kwint, dźwięki te nazywamy konsonansami doskonałymi. Są one podstawą akordów durowych. Jak widać kolejne grupy harmonicznym ( $k = 5, 10, 15$ ) oraz ( $k = 7, 14, 21$ ) są oddalone od tonu podstawowego odpowiednio o



**Fig. 320 Images of the base tones (*a*) and the first three overtones (*b*, *c*, *d*).**

Rysunek 1: Fale stojące dla struny. Wg Rainer Radok, Acoustics  
<http://kr.cs.ait.ac.th/~radok/physics/i4.htm>



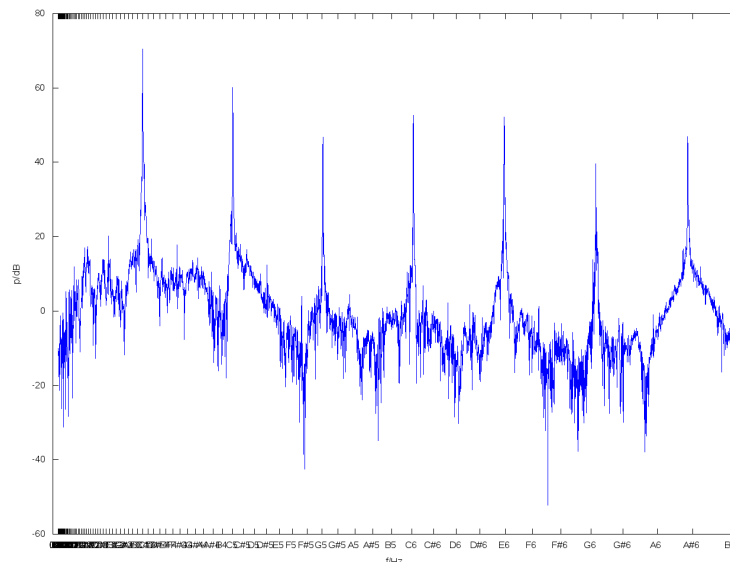
**FIGURE 22-4**

The Piano keyboard. The keyboard of the piano is a *logarithmic* frequency scale, with the fundamental frequency doubling after every seven white keys. These white keys are the notes: *A*, *B*, *C*, *D*, *E*, *F* and *G*.

Rysunek 2: Podział dźwięków na oktawy. Wg The Scientist and Engineer's Guide to Digital Signal Processing  
 By Steven W. Smith, Ph.D.

<http://www.dspguide.com/ch22/2.htm>

pewną ilość oktav i tercję bądź septymę. Są to tzw konsonanse niedoskonałe. Już starożytni zauważyli że ładnie współbrzmia (con-sonans - stąd konsonans) dźwięki, których częstotliwości stanowią wielokrotności. Opisuje to zjawisko tzw skala naturalna, w której interwały opisane są przez stosunki częstości postaci  $p/q$ , gdzie  $p$  i  $q$  to małe liczby naturalne (Wiki). Niestety skonstruowanie takiej skali, w której wszystkie częstości dźwięków byłyby swoimi wielokrotnościami jest niemożliwe, ponieważ wtedy różnice pomiędzy kolejnymi dźwiękami stają się zbyt duże. Z uwagi na to w wieku XIX zaproponowano skalę równomiernie temperowaną, która wprowadza niewielkie odstrojenia dla kwint i trochę większe dla tercji i septym, za to stosunek częstości sąsiednich dźwięków jest zawsze stały i równy  $\sqrt[12]{2}$ .



Rysunek 3: Widmo mocy dźwięku  $C_4$  pianina.

Przykład widma mocy obejmującego ton podstawowy i sześć pierwszych wyższych harmonicznych przedstawiony jest na rys. ???. Oś pozioma jest w skali liniowej żeby podkreślić fakt iż skala równomiernie temperowana ma charakter logarytmiczny (różnice częstotliwości między kolejnymi dźwiękami zależą od częstotliwości). Tonem podstawowym jest ton  $C_4$ . Jak widać kolejne harmoniczne przypadają (w przybliżeniu) na dźwięki (w nawiasach podano interwały:  $C_5$ (oktawa),  $G_5$ (kwinta),  $C_6$ (oktawa),  $E_6$ (tercja),  $G_6$ (kwinta),  $A\#_6$ (septyma)). W praktyce o barwie dźwięku decyduje pierwszych 10-12 harmonicznych. W przedziałach pomiędzy harmonicznymi widoczne jest widmo o niezerowej mocy. Jego źródła są dwojakie. Pierwszym źródłem są wyższe częstości (nieharmoniczne) - tzw alikwoty. W pierwszym przybliżeniu teorii alikwoty nie występują. To one decydują o barwie dźwięku: dla dwóch różnych instrumentów rozkład mocy alikwot będzie różny.

Wszystkie te dźwięki zagrane jednocześnie z tonem podstawowym będą ładnie współbrzmieć, tworząc konsonans (w przeciwieństwie do dysonansu który ładnie nie brzmi). Przykładowo trójdźwięk toniczny w tonacji  $C$  składa się z dźwięków  $C$ ,  $E$  i  $G$  a więc tonów oddalonych odpowiednio o tercję i kwintę od tonu podstawowego. Widać że te interwały nie są przypadkowe a wynikają z lokalizacji częstości harmonicznych drgającej struny. Konsonanse czyli współbrzmienia są podstawą harmonii - nauki traktującej o łączeniu dźwięków.

## Wykonanie ćwiczenia

Środowisko sugerowane: *Octave*.

Dane:

- <http://www.if.pw.edu.pl/~buchner/students/chord.wav> - akord w formacie WAV,
- <http://www.if.pw.edu.pl/~buchner/students/chord.bin> - akord w formacie BIN (Octave). lub

- <http://www.if.pw.edu.pl/~buchner/students/melodia1.wav> - melodia w formacie WAV, oraz
- <http://www.if.pw.edu.pl/~buchner/students/ex1.m> - w formacie Matlab/Octave,
- <http://www.if.pw.edu.pl/~buchner/students/tones.m> - w formacie Matlab/Octave,

Załaduj plik wav z dysku funkcją `wavread` lub `load` (jeśli nie ma `wavread`). Wykonaj wykres w funkcji czasu. Oceń stacjonarność sygnału (melodia czy akord). Wyznacz dla całego sygnału lub dla odpowiednich części transformatę Fouriera np z oknem Hamminga (lub dowolnym ale świadomie wybranym) a następnie widmo mocy. Wykonaj wykres widma mocy w jednostkach dB/Hz, dla przedziału 16Hz - 4kHz. Następnie zidentyfikuj akord lub melodię nagrane w pliku (= podaj wszystkie dźwięki wchodzące w skład akordu/melodii). Sugerowana metoda postępowania: obetnij wykres widma mocy (ale nie samo widmo) w pionie i w poziomie tak aby zostały tam tylko najwyższe piki w skali podwójnie logarytmicznej. Następnie dane zwracane przez funkcję `tones` wykorzystaj jako źródło położenia ticków oraz etykiet poziomej osi. Sposób użycia podany jest na końcu skryptu `tones.m`. Można także wykorzystać kod funkcji `notes` i napisać metodę która bierze częstość oraz częstość podstawową i zwraca najbliższy ton ze skali równomiernie temperowanej oraz ewentualnie odstrojenie do niego w jednostkach bezwzględnych (Hz) i względnych (podzielone przez samą częstość albo wyrażone w % interwału). Następnie zrób metodę która zidentyfikuje piki a następnie ich centralne częstości zmienić w nazwy półtonów. W przypadku pracy nad melodią trzeba zacząć od metody która zidentyfikuje chwile rozpoczęcia kolejnych dźwięków - dobrze widoczne na wykresie w funkcji czasu.

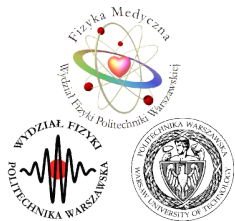
Procedury do użycia: `wavread`, `fft`, `hamming`, `conj`, `log10`, `plot`, `loglog`, `xlabel`, `ylabel`, `find`, `power`.

#### Uwagi:

1. Można korzystać ze źródeł w sieci pod warunkiem podania wszystkich źródeł.
2. Do zamiany obiektu `cell` na `string` służy instrukcja `str = cell2str( cell )`
3. Częstości zmierzone mogą różnić się nieco od częstości teoretycznych ponieważ instrument nie był idealnie nastrojony.
4. Uczciwość badacza nakazuje nie odsłuchiwać pliku melodii przed identyfikacją ;-).
5. Każdy dźwięk akordu składa się z tonu podstawowego oraz bogatej kolekcji alikwot (tonów wyższych), których obecność wynika z tego że widmo Fouriera pojedynczego dźwięku nie jest prążkiem.
6. Alikwoty mają częstości wyższe od częstości tonu podstawowego (czytaj: należy zacząć analizę od częstości najniższych).
7. Wśród alikwot pojedynczego dźwięku wyodrębnione są (na tle tła typu  $1/f$ ) prążki wyższych harmonicznych tonu podstawowego.
8. Wyższe dźwięki akordu mogą pokrywać się z alikwotami tonów niższych - na tym polega zjawisko współbrzmienia - asonansu (=dlatego jest ładnie).
9. Dźwięki należące do akordu powinny mieć natężenia większe niż natężenia w obrębie alikwot. Ale to nie musi być regułą.
10. Wskazówka. Akord jest w tonacji dur. Można odnaleźć w sieci wszystkie akordy durowe, wyznaczyć ich częstości dla każdej oktawy i porównać ich widmo teoretyczne z widmem otrzymanym. Na przykład można policzyć korelację widma teoretycznego z widmem doświadczalnym. W praktyce widmo teoretyczne lepiej w takim razie opisać jako krzywe Gaussa o pewnej szerokości.
11. Dźwięków w akordzie jest 6, mogą się pojawiać dźwięki wyższe o oktawę lub kilka oktaw od tonu podstawowego lub od dowolnego z tonów trójdźwięku durowego.

**Treść i forma sprawozdania:** forma elektroniczna (Word, TeX, HTML, pdf, ps, ...), sprawozdanie ma zawierać nazwę próbki, widmo mocy, opis metody identyfikacji częstości oraz zidentyfikowany akord: 6 tonów lub melodię zapisaną nazwami półtonów. Do sprawozdania należy dołączyć komplet wykorzystywanych skryptów Matlaba lub zapis sesji jeśli ktoś pracował interaktywnie.

whos-wyświetlanie wszystkich zmiennych z obszaru roboczego clear-czyszczenie obszaru roboczego



Instrukcja

Analiza Sygnału  
w Dziedzinie  
Czasu i Częstości  
dr Teodor Buchner

POLITECHNIKA WARSZAWSKA  
WYDZIAŁ FIZYKI  
PRACOWNIA FIZYKI UKŁADU KRĄŻENIA CZŁOWIEKA  
[efizyka.if.pw.edu.pl/twiki/bin/view/ACC/](http://efizyka.if.pw.edu.pl/twiki/bin/view/ACC/)



KAPITAŁ LUDZKI  
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



PRZYGOTOWANIE I REALIZACJA SPECJALNOŚCI WSPÓLFINANSOWANE ZE ŚRODKÓW UNII EUROPEJSKIEJ W RAMACH EUROPEJSKIEGO FUNDUSZU SPOŁECZNEGO

## Ćwiczenie 2 *FFT-logia*

### Wprowadzenie

Patrz Wykład. Ćwiczenie ma na celu zapoznanie z podstawowymi własnościami transformaty Fouriera w krótkim oknie czasowym oraz nabycie biegłości w posługiwaniu się środowiskiem Matlab/Octave.

### Wykonanie ćwiczenia

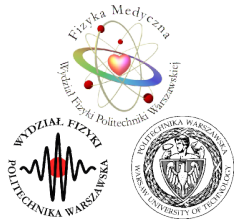
**Środowisko sugerowane:** *Octave*.

Przygotuj skrypt który zilustruje następujące zagadnienia

1. Twierdzenie o próbkowaniu. Porównanie sygnałów w funkcji czasu oraz widm (obrazki w dwóch kolumnach):
  - (a) Sygnału nieskończonego (wykres teoretyczny),
  - (b) Funkcji okna prostokątnego (należy dobrać okno krótkie, obejmujące co najwyżej kilka okresów sygnału)
  - (c) Sygnału skończonego czyli iloczynu sygnału nieskończonego i funkcji okna prostokątnego
  - (d) Funkcji okna nieprostokątnego (należy dobrać okno krótkie, obejmujące co najwyżej kilka okresów sygnału)
  - (e) Sygnału skończonego wygładzonego oknem czyli iloczynu sygnału nieskończonego i funkcji okna nieprostokątnego
2. Aliasing. Widmo sygnału próbkowanego poniżej i powyżej częstotliwości Nyquista. Jako sygnał można wykorzystać jeden z fragmentów muzycznych z poprzedniego ćwiczenia. Można także wziąć swoją ulubioną piosenkę w formacie wav. Częstość próbkowania jest zapisana w nagłówku pliku wav i zwracana przez funkcję `wavread`. Najlepiej jest zacząć od próbkowania powyżej częstotliwości Nyquista a następnie wykonać kilkukrotnie decymację (usuwanie co drugiej próbki), za każdym razem wyświetlając wykres sygnału i widma.
3. Zero padding - do 4x długość sygnału (do analizy bierzemy albo sygnał teoretyczny albo fragment muzyczny).
4. Upsampling - do 4 x krotnie (brakujące próbki wypełniamy zerami.) (do analizy bierzemy albo sygnał teoretyczny albo fragment muzyczny)

W sieci jest wiele dobrych tutoriali Matlab/Octave, na przykład:

- <http://wiki.aims.ac.za/mediawiki/index.php/Octave>
- <http://www.phon.ucl.ac.uk/courses/spsci/matlab/>



Instrukcja

Analiza Sygnału  
w Dziedzinie  
Czasu i Częstotliwości  
dr Teodor Buchner

POLITECHNIKA WARSZAWSKA  
WYDZIAŁ FIZYKI

PRACOWNIA FIZYKI UKŁADU KRĄŻENIA CZŁOWIEKA  
[efizyka.if.pw.edu.pl/twiki/bin/view/ACC/](http://efizyka.if.pw.edu.pl/twiki/bin/view/ACC/)



KAPITAŁ LUDZKI  
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



PRZYGOTOWANIE I REALIZACJA SPECJALNOŚCI WSPÓLFINANSOWANE ZE ŚRODKÓW UNII EUROPEJSKIEJ W RAMACH EUROPEJSKIEGO FUNDUSZU SPOŁECZNEGO

## Ćwiczenie 3 Czas i częstotliwość

### Wprowadzenie

Patrz Wykład. Ćwiczenie ma na celu porównanie transformaty Fouriera w krótkim oknie czasowym (STFT) i transformat biliniowych - Wigner-Ville i innych transformat z klasy Cohena.

### Wykonanie ćwiczenia

**Środowisko sugerowane:** *Octave*.

Uwaga: wszystkie **pliki** *XXXX.m* wzmiankowane poniżej można znaleźć pod adresem <http://www.if.pw.edu.pl/~buchner/students/XXXX.m>

Przygotuj skrypt który wykona następujące czynności:

- Wygeneruje sygnał o długości  $N = 512$  i częstotliwości próbkowania  $f_s = 500\text{Hz}$ : świergot liniowy o częstotliwości początkowej  $f_1 = 20\text{Hz}$  i końcowej  $f_2 = 100\text{Hz}$ . Świergot generuje funkcja *chirp* której argumenty to czas (należy go zdefiniować jako wektor - np  $tn = (1:N)/fs$ ), częstość początkowa, czas w którym ma zostać osiągnięta częstość końcowa (czyli  $N/fs$ ) oraz częstość końcowa. Jest to wyjaśnione pod *help chirp*.
- Wygeneruje transformatę Wigner-Ville z sygnału. Ponieważ transformata VW wymaga funkcji analitycznej, przed transformatą VW trzeba wykonać transformatę Hilberta sygnału. Służy do tego funkcja *hilbert*. Transformaty z klasy Cohena zaimplementowane są w pliku *cohen.m*, który dodatkowo wymaga plików kerneli (jąder) (por. wykład). Obecnie zaimplementowany jest kernel Choi-Williams *choi.m*. Dodatkowo wykorzystywana jest funkcja dwuwymiarowego spłotu: *conv2.m* (powinna być standardowo w dystrybucji Octave) oraz autokorelacji *int\_autocorr.m*. Funkcja *cohen* bierze jako argumenty fcję analityczną, częstość próbkowania oraz argument który określa jaki kernel ma być użyty. Argument *VW* daje kernel Wigner-Ville. Funkcja zwraca transformatę jako funkcję rzeczywistą dwuwymiarową i dodatkowo wektor czasu i wektor częstości, których można użyć do opisywania osi wykresu. Ponieważ transformata W-V może dawać wartości ujemne, przed wykreśleniem należy wziąć wartość bezwzględną. Należy wyświetlić moduł transformaty oraz logarytm modułu. Do rysowania 3D służy wiele funkcji, najprostsza to *imagesc(t,f,<tu Twoja funkcja 2-D>);axis xy;*
- Wygeneruje spektrogram - czyli transformatę Fouriera w krótkim oknie. Do generacji spektrogramu można użyć metody *spectog* zaimplementowanej w pliku *spectog.m*. Argumenty funkcji *spectog* to: sygnał, długość okna FFT:  $nfft$ , częstość próbkowania oraz długość zachodzenia (overlapping). Określa ona czy sąsiednie okna mają zachodzić na siebie. Standardowa wartość dla parametru overlapping to  $nfft/2$ . Rysujemy w ten sam sposób. Należy porównać wykresy dla różnych wartości  $nfft$ : 16, 32, 64, 128 i 256 a także 512 dla sygnału dwukrotnie dłuższego: o długości  $N = 1024$  lub czterokrotnie dłuższego  $N = 2048$ . i omówić uzyskany wynik z punktu widzenia zasady nieoznaczoności Heisenberga (w wersji teorisygnałowej - patrz wykład).



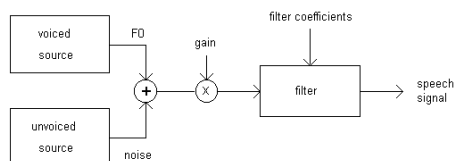
- Natępnie należy wybrać określoną długość okna FFT - na przykład  $nfft = 64$ , a następnie pokazać obok siebie transformatę logarytmu VW dla sygnału o długości  $N = 512$  oraz STFT. W tak przygotowanych warunkach należy:
  - zmienić częstość końcową świergotu do  $f_2 = 200Hz$  a następnie  $f_2 = 500Hz$  (można także wyświetlić pośrednie: 220,250,270,300,400 itp.). Następnie skomentować wynik (w sprawozdaniu). Skomentować=zrozumieć skąd się wziął.
  - Wyznaczyć widmo dla sumy dwóch świergotów zmieniających się równolegle: jednego który zmienia się od  $f_1 = 20$  do  $f_2 = 200$  i drugiego który zmienia się od  $f'_1 = 2 \cdot f_1$  do  $f'_2 = f_2 + f_1$ .
  - Wyznaczyć widmo dla sumy dwóch świergotów zmieniających się nierównolegle: jednego który zmienia się od  $f_1 = 20$  do  $f_2 = 200$  i drugiego który zmienia się od  $f'_1 = 2 \cdot f_1$  do  $f'_2 = 2 \cdot f_2$ .

Wszystkie otrzymane wyniki skomentować.

## Ćwiczenie 4 *Analiza parametryczna, synteza i kompresja mowy*

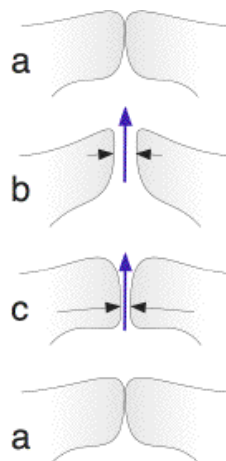
### Wprowadzenie

Cwiczenie ma na celu wprowadzenie w analizę parametryczną sygnału na przykładzie analizy sygnału mowy. Podstawowym modelem wytwarzania mowy jest model źródło-filtr (Fant, G. (1960). *Acoustic Theory of Speech Production*. Mouton & Co, The Hague, Netherlands.). W modelu tym źródłem dźwięku są struny głosowe, zaś głośnia (trakt głosowy) filtruje ton podstawowy i nadaje mu barwę.



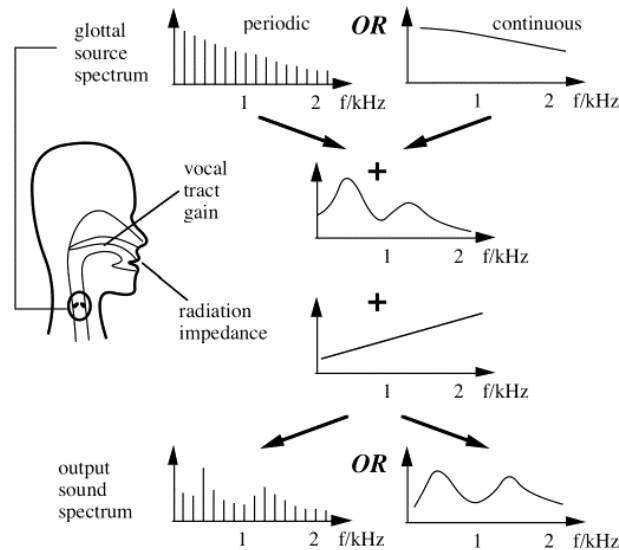
Rysunek 4: Schemat modelu źródło-filtr.

Struny głosowe mają dwa tryby pracy: dźwięczny (głoski dźwięczne) i bezdźwięczny (głoski bezdźwięczne). W trybie dźwięcznym struny są częściowo zamknięte a strumień powietrza przepływający przez nie podlega modulacji. Modulacja ta nie ma charakteru czynnego ponieważ mięśnie gładkie nie są w stanie wytworzyć drgań o częstotliwości fali głosowej.



Rysunek 5: Schemat działania strun głosowych

Mechanika wytwarzania dźwięku w trybie dźwięcznym polega na tym że struny zbliżają się do siebie na pewną odległość. Ciśnienie powietrza wypychanego z płuc powoduje ich otwarcie, co wywołuje przepływ powietrza. Ten przepływ na mocy prawa Bernoulliego powoduje iż na struny działa siła, która z uwagi na ich elastyczność zbliża je do siebie i cykl się zamyka. Powoduje to powstanie w strumieniu powietrza następujących kolejno po sobie zmian ciśnienia akustycznego (zgęszczeń i rozrzedzeń). Ich częstość - tzw częstość podstawowa wynosi od 100 do 400 Hz dla mowy zaś od 60 Hz do 1500 Hz dla śpiewu. Ich charakterystyka przypomina bardziej ciąg delt Diraca niż sinusoidę. Regulacja działania strun przez układ nerwowy polega na regulacji ich odległości początkowej oraz własności elastycznych określających podatność na siłę wywołaną przepływem. W trybie bezdźwięcznym struny głosowe nie wykonują pracy i strumień powietrza pozostaje niezmodulowany. Tak spreparowany strumień powietrza o ciągłej (głoski bezdźwięczne) lub dyskretnej (głoski dźwięczne) charakterystyce widmowej trafia do traktu głosowego. Trakt głosowy stanowi rezonator. Sterowanie nim polega na nadawaniu mu określonego kształtu, przez co określone częstości są wzmacniane lub wygaszane.



Rysunek 6: Schemat wytwarzania sygnału mowy

Struktura modelu źródło-filtr (rys. ??) odpowiada opisanej powyżej strukturze emisji głosu. Dla głosek dźwięcznych (a,e,i,o,u) źródłem jest generator tonu podstawowego wytwarzający ciąg delt Diraca o częstotliwości podstawowej. Dla głosek bezdźwięcznych (s,sz,c,cz,h) źródłem jest szum szerokopasmowy. Następnie sygnał trafia do filtra traktu głosowego który nadaje mu określoną charakterystykę częstotliwościową.

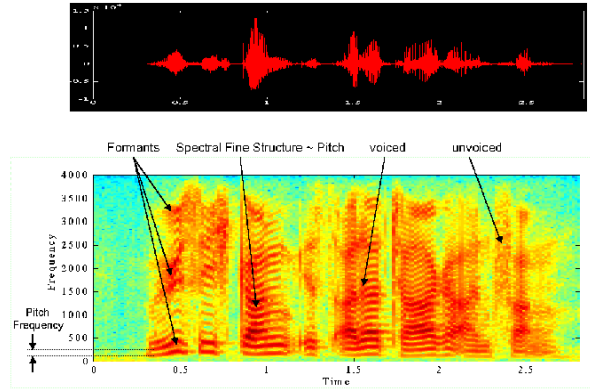
W sygnale mowy wyróżniamy tzw formanty - widmo głosek składa się z określonych częstości, co pozwala na ich identyfikację (rozpoznawanie mowy)

W odniesieniu do sygnału mowy w akustyce rozróżnia się cztery podstawowe techniki: analizę, rozpoznawanie, syntezę i związaną z nimi kompresję.

Do celów analizy mowę dzieli się na ramki o długości 30 ms (dynamika traktu głosowego nie pozwala na szybsze zmiany), co przy stosowanym standardowo próbkowaniu 8 kHz daje ramkę o długości  $N=240$  próbek. Odstęp między ramkami wynosi  $M=180$  próbek, a więc sąsiednie ramki nachodzą na siebie.

Analiza sygnału mowy (opisany poniżej jest podstawowy algorytm LPC-10) polega na:

1. Określeniu czy głoska jest dźwięczna czy bezdźwięczna
2. Określeniu częstości tonu podstawowego dla głosek dźwięcznych
3. Określeniu parametrów filtra który jest modelem traktu głosowego.
4. Określeniu wzmocnienia



Rysunek 7: Schemat wytwarzania sygnału mowy

Zadania 1 i 2 można zrealizować przez wyznaczenie i analizę autokorelacji sygnału:

$$r(k) = \sum_{n=0}^{N-1-k} s(n)s(n+k), \quad k = 0, 1, \dots, N-1 \quad (8)$$

Analizowany fragment mowy zostaje uznany za dźwięczny jeśli w przedziale  $k \in [20-160]$  (częstości  $50-400\text{Hz}$ ) istnieje maksimum autokorelacji mające przynajmniej wysokość 0.3 maksimum autokorelacji  $r(0)$ . Położenie tego maksimum  $k_{max}$  odpowiada częstości tonu podstawowego (odstępowi pomiędzy kolejnymi deltami Diraca, mierzonemu w liczbie próbek). Druga metoda polega na użyciu tzw cepstrum. Cepstrum definiowane jest jako odwrotna tr Fouriera z logarytmu widma mocy sygnału. Pozwala ono na oddzielenie dwóch transmitancji w dziedzinie częstości: jeśli sygnał jest przetwarzany przez dwa kolejne filtry, użycie logorytmu powoduje iż otrzymane cepstrum stanowi sumę dwóch składowych w dziedzinie czasu. Obwiednia widma pojawia się w okolicy indeksu 0, zaś prążki widma powodują powstanie pików o malejącej amplitudzie w punktach odległych o wielokrotność tonu podstawowego.

Zadanie 3 i 4 można zrealizować przez: Dobór filtra autoregresyjnego (predyktor jednokrokový) rzędu  $p = 10$ , który na podstawie poprzednich próbek, mnożonych przez współczynniki filtra  $[a_1, \dots, a_{10}]$  „zgaduje” próbkę następną. Można pokazać iż średniokwadratowy błąd predykcji jest minimalny jeśli wektor współczynników filtra jest rozwiązaniem równania macierzowego dla  $p = 10$ :

$$\mathbf{a} = -\mathbf{R}^{-1} \mathbf{r}, \quad \text{gdzie} \quad (9)$$

$$\mathbf{a} = [a_1, a_2, \dots, a_p]^T, \quad (10)$$

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & \dots & r(p-1) \\ r(1) & r(0) & \dots & r(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(p-1) & r(p-2) & \dots & r(0) \end{bmatrix} \quad (11)$$

Wzmocnienie określa się ze wzoru:

$$G = r(0) + \sum_{k=1}^p a_k \cdot r(k) \quad (12)$$

Wyznaczenie parametrów częstotliwościowych traktu głosowego polega na określeniu transmitancji filtra traktu głosowego (funkcja *freqz*). Pojawienie się określonych formantów można wykrywać, i w ten sposób rozpoznaje się kolejne głoski.

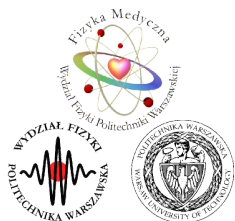
Kompresja mowy jest możliwa dzięki temu że dla pojedynczej ramki przesyłane są jedynie parametry filtra, częstość podstawowa (równa 0 dla głosek(ramek)bezdźwięcznych) oraz wzmocnienie, co daje kompresję 27-krotną. Synteza mowy polega na odtworzeniu dźwięku na podstawie opisanych powyżej danych.

# Wykonanie ćwiczenia

**Środowisko sugerowane:** *Octave*.

Przygotuj skrypt który wykona następujące czynności:

- Dla dowolnego sygnału mowy (mono, prókowanie 8 kHz) określi dla każdej ramki:
  - dźwięczność lub bezdźwięczność ramki przez analizę autokorelacji lub cepstrum
  - Parametry filtra traktu głosowego
  - Wykres czas-częstość (transmitancja filtra traktu głosowego dla kolejnych ramek)
  - Syntezę (dla chetnych)



Instrukcja

Analiza Sygnału  
w Dziedzinie  
Czasu i Częstotliwości  
dr Teodor Buchner

POLITECHNIKA WARSZAWSKA  
WYDZIAŁ FIZYKI

PRACOWNIA FIZYKI UKŁADU KRĄŻENIA CZŁOWIEKA  
[efizyka.if.pw.edu.pl/twiki/bin/view/ACC/](http://efizyka.if.pw.edu.pl/twiki/bin/view/ACC/)



KAPITAŁ LUDZKI  
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



PRZYGOTOWANIE I REALIZACJA SPECJALNOŚCI WSPÓLFINANSOWANE ZE ŚRODKÓW UNII EUROPEJSKIEJ W RAMACH EUROPEJSKIEGO FUNDUSZU SPOŁECZNEGO

## Ćwiczenie 5 *Transformata falkowa*

### Wprowadzenie

Ćwiczenie ma na celu wprowadzenie w analizę czas-częstotliwość przy użyciu transformaty falkowej. Jako przykład sygnału niestacjonarnego wykorzystamy sygnał mowy oraz sygnał syntetyczny.

Jednym z podstawowych zagadnień w analizie, rozpoznawaniu i syntezy mowy ludzkiej jest rozpoznanie tzw. formantów - struktur czasowo-częstotliwościowych odpowiadających konkretnym głoskom. Struktury te widoczne są w wynikach transformacji czas-częstotliwość otrzymanej w ćwiczeniu poprzednim. Analizy formantów dokonuje się także w przypadku analizy dźwięków wydawanych przez zwierzęta (nietoperze, walenie). Analiza tego typu jest również użyteczna w zagadnieniach inżynierskich - por. Białasiewicz - gdzie pojawienie się drgań o niskich częstotliwościach jest interpretowane jako efekt zmęczenia badanych mechanizmów lub problem konstrukcyjny, któremu należy zapobiegać przez modyfikację drgających mechanizmów.

Transformata falkowa, podobnie jak inne metody analizy czas-częstotliwość bywa często wykorzystywana do tego typu analiz. Zapoznamy się tutaj z podejściem zorientowanym numerycznie, które pozwala na przyspieszenie obliczeń. Dość wspomnieć że istnieją implementacje dyskretnej transformaty falkowej (DWT), które umożliwiają jej wyznaczanie w czasie rzeczywistym, co pozwala na wykorzystanie jej wyników w sterownikach przemysłowych.

Układ liniowy (czwórnik) możemy traktować jak filtr w dziedzinie częstotliwości. Filtracja w dziedzinie częstotliwości jest równoważna operacji splotu w dziedzinie czasu. Mnożenie sygnału wejściowego (np. harmonicznego) przez transmitancję daje odpowiedź układu w dziedzinie częstotliwości. Analogicznie odpowiedź układu w funkcji czasu można uzyskać przez operację splotu sygnału wejściowego z funkcją odpowiedzi na skok jednostkowy, która stanowi odwrotną transformatę Fouriera funkcji transmitancji.

Operację splotu można postrzegać jako iloczyn skalarny badanego sygnału i danej funkcji bazowej. Zatem wyjście układu liniowego (filtra) złożone jest z kolejno wyznaczonych iloczynów skalarnych, gdzie funkcją bazową jest odpowiedź na skok jednostkowy. Wynikiem są współczynniki rozwinięcia badanego sygnału w bazie nieskończenie wielu kopii funkcji  $h$ , różniących się argumentem, który jest przesunięty za każdym razem o 1 próbkę. Taka operacja jest możliwa jedynie skończonej długości nośnika funkcji bazowych. Przy odpowiedniej konstrukcji funkcje takie dla przesuniętych wartości argumentu są ortogonalne.

Opisana powyżej operacja jest istotą transformaty falkowej, przy czym powtarza się ją najpierw dla różnych lokalizacji w czasie "środku" funkcji bazowej, a następnie, co będzie opisane poniżej, dla różnych skal czasowych.

W transformacie falkowej sygnał przepuszczany jest na każdym poziomie przez parę filtrów, które mają własność sprzężonych filtrów zwierciadlanych (quadrature mirror filters - QMF): suma mocy sygnału na wyjściu filtrów jest równa mocy sygnału na wejściu. Zarazem suma kwadratów transmitancji jest równa 1. Jeden z filtrów ma charakter dolnoprzepustowy (low pass - LP) a drugi - górnoprzepustowy (high pass - HP). Sygnał na wyjściu filtra HP to współczynniki rozwinięcia w bazie falek (jest to zarazem wynik transformaty falkowej na danym poziomie), zaś sygnał na wyjściu filtra LP to współczynniki rozwinięcia w bazie funkcji skalujących. Następnie obie grupy współczynników poddane są tzw. decymacji - używana jest co druga wartość. W ten sposób analizowane pasmo częstotliwości przesuwane jest o połowę swojej wartości w dół. Następnie operacja filtracji

jest powtórzona; tym razem na wejściu obu filtrów jest wektor współczynników, które wyszły z filtra dolnoprzepustowego. W ten sposób na wyjściu filtra HP otrzymujemy kolejne współczynniki rozwinięcia w bazie falek przeskalowanych w dziedzinie czasu (i zarazem częstości). Po wielokrotnej realizacji tych operacji otrzymujemy tzw skalogram: zbiór współczynników transformaty falkowej odpowiadający rozwinięciom w bazie falek o różnych skalach. Technika opisana powyżej realizuje rozpięcie na siatce diadycznej, na której skala zmienia się o  $\frac{1}{2}$ . W związku z tym analizowany sygnał jest za każdym razem  $2x$  krótszy, zaś procedurę kończymy po wykonaniu analizy w zadanej liczbie skal, przy czym liczba skal jest albo zadana jako parametr albo określona przez długość sygnału wejściowego - jako  $\log_2 N$ , gdzie  $N$  - długość sygnału.

Opisaną powyżej operację można obrócić, co realizuje tzw odwrotną transformatę falkową. Różnica jest taka iż w miejsce filtrów LP i HP analizy falkowej wstawiane są filtry syntezy o innych współczynnikach, zaś po zmianie skali sygnał jest uzupełniany zerami, co stanowi odwrotność decymacji.

Jeśli przy syntezie niektóre skale zostaną pominięte, sygnał wyjściowy zostaje poddany filtracji. Tą samą drogą można dokonywać kompresji sygnału. Operacja taka wykorzystywana jest w standardzie JPEG-2000.

Filtracja w wielu skalach w praktyce numerycznej wykorzystuje wielokrotnie te same filtry, przy zmieniającym się sygnale wejściowym. Formalnie można tę operację przedstawić jako filtrację przez zespół filtrów.

Każda falka ma swój zakres częstości - w istocie obejmujący pewne pasmo częstotliwości. W związku z tym można formalnie przekształcić skalogram w diagram czas-częstość, przypisując każdej ze skal częstość centralną falki o danej skali.

## Wykonanie ćwiczenia

**Środowisko sugerowane:** *Octave*.

Przygotuj skrypt który wykona następujące czynności:

- Wygeneruje falkę podstawową typu Coiflet1

```
f = [ .038580777748 -.126969125396 -.077161555496 ...  
      .607491641386 .745687558934 .226584265197 ];
```

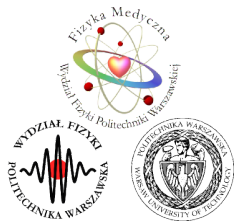
- Przeanalizuje dowolny fragment nietrywialnego sygnału o długości  $2^L$  (może być sygnał mowy z poprzedniego ćwiczenia) za pomocą dyskretniej transformaty falkowej, a więc:
  - Utworzy filtr zwierciadlany do falki Coiflet - dolnoprzepustowy przez operację:
 
$$y = -((-1)^{(1:\text{length}(x)}) ) \cdot x;$$
  - Dokona filtracji, a więc wykona splot cykliczny (por wykład) sygnału z filtrem falki (górnoprzepustowym) i filtrem zwierciadlanym do falki (dolnoprzepustowym)
  - *Uwaga* Splot stosowany w przykładzie z Zielińskiego prawdopodobnie nie jest cykliczny.
  - Dokona decymacji współczynników które stanowią wynik splotu dla obu filtrów
  - Zapamięta współczynniki filtracji filtrem górnoprzepustowym jako współczynniki rozwinięcia falkowego dla danej skali
  - Podstawi zdecydowane współczynniki filtracji filtrem dolnoprzepustowym jako sygnał dla analizy w następnej skali
  - Wykona powyższą operację maksymalną możliwą ilość razy (dla danej długości sygnału).
- Następnie należy wyznaczyć częstość centralną falki dla każdej skali
- Następnie należy narysować skalogram badanego sygnału, zaś oś skal wyskalować w Hz.
- *Wskazówka* Proponuję zacząć badanie od prostych sygnałów (sygnał harmoniczny, chirp) żeby było dobrze widać co się dzieje w wynikach.

- *Wskazówka* Można wybrać dowolną inną falę, byle nie db4 i o długości innej niż 4.
- *Wskazówka* Można posłużyć się pakietem WaveLab 850 do ściągnięcia z uniwersytetu Stanford. Interesujące skrypty kompatybilne z Octave znajdują się w katalogu

Wavelab850/Orthogonal

. Nie należy instalować tego pakietu przy pomocy dołączonego instalatora tylko dołączyć tę ścieżkę (i wszystkie których będzie potrzebował) do ścieżek Octave przez addpath(). Instalator wyklada się na różnicach między Matlabem i Octave więc jest średnio użyteczny. W katalogu Book są liczne przykłady.





Instrukcja

Analiza Sygnału  
w Dziedzinie  
Czasu i Częstości  
dr Teodor Buchner

POLITECHNIKA WARSZAWSKA  
WYDZIAŁ FIZYKI

PRACOWNIA FIZYKI UKŁADU KRĄŻENIA CZŁOWIEKA  
[efizyka.if.pw.edu.pl/twiki/bin/view/ACC/](http://efizyka.if.pw.edu.pl/twiki/bin/view/ACC/)



KAPITAŁ LUDZKI  
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



PRZYGOTOWANIE I REALIZACJA SPECJALNOŚCI WSPÓLFINANSOWANE ZE ŚRODKÓW UNII EUROPEJSKIEJ W RAMACH EUROPEJSKIEGO FUNDUSZU SPOŁECZNEGO

## Ćwiczenie 6 Przetwarzanie obrazów. Format graficzny JPEG i JPEG 2000

### Wprowadzenie

Transformata Fouriera i transformata falkowa może być z powodzeniem używana w algorytmach kompresji obrazów, dźwięków i filmów. Obejrzymy to na przykładzie standardów kompresji obrazów JPEG i JPEG-2000. Obydwa standardy oparte są o transformację z oryginalnej dziedzinie do dziedzinie częstości. Zaczynają się od podziału obrazu na bloki  $N \times M$ . Nie jest to bezwzględnie wymagane przez standardy, ponieważ algorytmy radzą sobie także bez podziału na bloki (w praktyce: do rozmiaru  $2^{32} \times 2^{32}$ ). Standardowo stosowany jest blok  $8 \times 8$ . Jeśli rozmiar obrazu nie jest wielokrotnością  $N \times M$ , obraz zostaje uzupełniony do najbliższej wielokrotności w pionie i w poziomie. Następnym krokiem jest dokonanie odpowiedniej transformacji (kosinusowej dla standardu JPEG lub falkowej dla standardu JPEG 2000) a następnym - kwantyzacja w dziedzinie częstości, której zadaniem jest właściwa kompresja obrazu. Mogą być stosowane różne algorytmy kwantyzacji - mniej lub bardziej agresywnie kompresujące obraz. Jeśli obraz składa się z wielu składowych, takich jak np kanały R, G, B,  $\alpha$  lub Y, U, V, albo w (najprostszym) przypadku obrazów czarno-białych stopnie szarości.

Do pomiaru jakości kompresji służy wielkość PSNR (peak signal to noise ratio) [?, ?], która ma sens błędu całkowitego i wyrażona jest w dB. PSNR zdefiniowana jest dla oryginału  $f$  i obrazu skompresowanego  $f'$  jako:

$$PSNR = 10 \cdot \log_{10} \frac{k^2}{MSE}, \text{ gdzie } k = (\text{liczba kolorów} - 1), \text{ zaś } MSE \text{ to:} \quad (13)$$

$$MSE = \frac{1}{N \cdot M} \sum_{i=1}^N \sum_{j=1}^M ([f(i, j) - f'(i, j)]^2) \quad (14)$$

Jak widać, MSE trafia do mianownika, zatem im wyższa wartość PSNR tym lepiej. PSNR można zmierzyć dla różnych stopni kompresji wyrażonych w bitach na piksel (bpp).

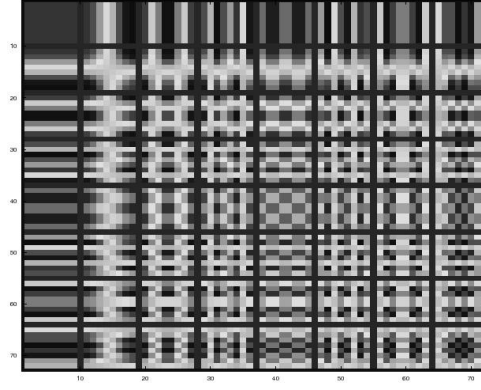
### Standard JPEG

Transformacją używaną w standardzie JPEG jest dwuwymiarowa transformacja cosinusowa. Jest ona złożeniem dwóch kolejnych jednowymiarowych transformat kosinusowych (DCT - discrete cosine transform). Najpierw każdy wiersz obrazu zastępuje się współczynnikami rozwinięcia w bazie kosinusów. Następnie każdą kolumnę tak sporządzonego obrazu zastępuje się współczynnikami rozwinięcia w bazie kosinusów. Numerycznie można zrealizować DCT za pomocą FFT. Formalnie transformacja dana jest wzorem (??) (za: [?]):

$$X_{DCT}(k, l) = \sum_{m=0}^{M-1} \left[ \sum_{n=0}^{N-1} x(m, n) \cdot \beta(l) \cdot \cos \left( \frac{\pi l}{N} (n + 1/2) \right) \right] \cdot \alpha(k) \cdot \cos \left( \frac{\pi k}{M} (m + 1/2) \right) \quad (15)$$

gdzie

$$\alpha(k) = \begin{cases} \sqrt{1/M}, & k = 0 \\ \sqrt{2/M}, & k = 1 \dots M - 1 \end{cases}, \beta(l) = \begin{cases} \sqrt{1/N}, & l = 0 \\ \sqrt{2/N}, & l = 1 \dots N - 1 \end{cases} \quad (16)$$



Rysunek 8: Częstotliwości przestrzenne

Transformację tę można także zapisać w postaci równania macierzowego (??), co jest bardzo wygodne dla zastosowań numerycznych:

$$Y_{DCT}(k, l) = \sum_{m=0}^{M-1} \left[ \sum_{n=0}^{N-1} x(m, n) \cdot b_N^*(l, n) \right] \cdot b_M^*(k, m) \quad (17)$$

$$Y_{M \times N} = B_M^* \cdot X_{M \times N} \cdot B_N^{*T} \quad (18)$$

gdzie  $b_P(k, m)$   $k, m \in [0, P-1]$  - funkcje bazowe - są zapisane w kolejnych wierszach macierzy  $B$ . Na przykład:

$$b_P(k, m) = \sqrt{\frac{1}{P}} \exp \left( j \frac{2\pi k}{P} m \right) \quad \text{dla FFT} \quad (19)$$

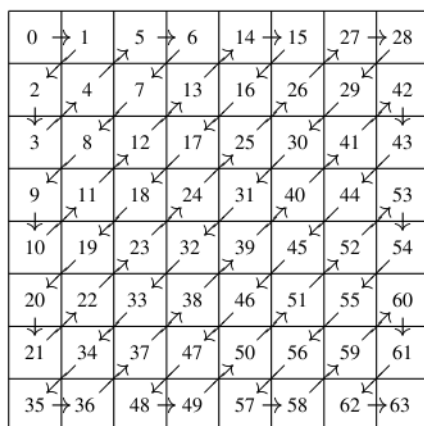
$$b_P(k, m) = \alpha(k) \cdot \cos \frac{\pi k}{P} (m + 1/2) \quad \text{dla DCT} \quad (20)$$

Rysunek ?? przedstawia coraz wyższe częstotliwości przestrzenne przedstawione jako macierze. Analizowaną macierz można przedstawić w postaci sumy macierzy z rysunku ?? wymnożonych przez współczynniki rozwinięcia (por [?], wzory 22.24-22.26).

Zauważmy że częstotliwości najniższe (najniższe wartości współczynników  $k, l$  we wzorze (??) ułożone są w lewym górnym rogu. Częstota rośnie w prawo i w dół. Transformacja (??) pozwala na filtrację przestrzenną, która polega na wymnożeniu macierzy współczynników rozwinięcia widmowego przez macierz transmitancji filtra. Obszerny materiał na ten temat podaje Zieliński [?].

W związku z lokalizacją częstotliwości przestrzennych, zamiana macierzy w wektor według wzoru przedstawionego na rysunku ?? szereguje współczynniki spektralne według rosnącej częstotliwości (czy raczej sumy częstotliwości).

Kolejny etap kompresji to kwantyzacja, która polega na podzieleniu macierzy  $Y$  przez macierz o tym samym wymiarze, zawierającą liczby rosnące zgodnie z zygzakiem, asymetryczne (tzw standardowa macierz luminancji - otrzymana doświadczalnie) lub symetryczne (tzw macierz Hilberta) względem głównej przekątnej. Wyrazy macierzy otrzymanej po podzieleniu zostają zaokrąglone w dół do najbliższej liczby całkowitej. Miara stopnia kompresji jest liczba zer, które pojawiają się w wynikowej macierzy, zaś jej współczynniki po zamianie w wektor danych (wg zygzaka z rys. ??) zostają poddane kompresji typu Huffmana, zapewniającej minimalną entropię przesyłanego ciągu symboli (kolejna forma kompresji) mającej na celu przypisanie ciągom liczb, występujących w badanym ciągu danych, etykiet ze słownika, przy czym etykieta jest tym krótsza im częściej pojawia się dany symbol czy słowo. Szczegóły podaje Zieliński [?] w rozdziale 11, są one także opisane w literaturze z algorytmiki.



Rysunek 9: Technika zamiany macierzy w wektor.

## Standard JPEG 2000

Wadą JPG jest występująca na skutek agresywnej kwantyzacji pikseloza, pokazująca rozmiar bloku (np  $8 \times 8$ ). Tej wady pozbawiona jest metoda JPEG 2000 dająca bardzo dobre współczynniki kompresji bez widocznej zmiany jakości obrazu. Wadą tej metody jest spore zapotrzebowanie na moc obliczeniową oraz nie do końca rozwiązane kwestie zależności od istniejących patentów (por. <http://www.answers.com/topic/jpeg-2000>). Dla standardu JPEG2000 (w przeciwieństwie do JPEG) im większy rozmiar bloku tym lepiej dla jakości obrazu. Standard JPEG 2000 różni się od poprzednika zarówno konstrukcyjnie jak i osiąganym stopniem kompresji, wyrażonym przez wysokie wartości PSNR.

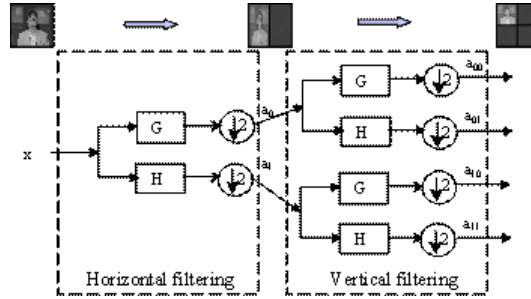
Kompresja JPEG2000 rozpoczyna się także od podziału obrazu na bloki. Następnie dla każdego bloku jest wykonywana zwykła jednowymiarowa filtracja przez kaskadę zwierciadlanych filtrów kwadraturowych (QMF), podobnie jak w zwykłej transformacji falkowej. Filtracja wykonywana jest najpierw dla każdego wiersza obrazu oryginalnego a potem dla kolumn. Wynikiem każdej transformacji falkowej są dwa obrazy: jeden jest wynikiem filtracji górnoprzepustowej (HP) - ten sygnał można określić jako detale (D) (por. [?] rys 22.32), zaś drugi - tzw aproksymata (A) - wynikiem filtracji dolnoprzepustowej (LP). Ponieważ filtracja wykonywana jest w dwóch prostopadłych kierunkach, z jednego obrazu otrzymujemy po pierwszym kierunku obrazy LP i HP (A i D), zaś po drugim kierunku aż cztery, ponieważ każdy z obrazów ponownie wchodzi na parę filtrów QMF. Mamy zatem cztery obrazy. Z LP powstaje LP-LP i LP-HP, zaś z HP: HP-HP i HP-LP. Elementem transformacji jest decymacja. Po filtracji wierszy decymacja prowadzi do wyboru co drugiego współczynnika, czyli co drugiej kolumny, zatem obraz jest dwukrotnie węższy. Następnie po filtracji kolumn decymacja powoduje iż cztery obrazy mają dwukrotnie mniejszą zarówno wysokość jak i szerokość. Obrazy D-D (HP-HP), D-A (HP-LP) i A-D (LP-HP) są zapisywane "na boku", natomiast obraz A-A (LP-LP) ponownie wchodzi na filtry QMF następnego stopnia transformacji falkowej. Znowu, z uwagi na dokonaną decymację możemy użyć tych samych filtrów.

Współczynniki rozwinięcia falkowego kolejnych stopni można zapisać w określonych regionach zamiast oryginalnego obrazu. Schemat pokazany jest na rysunku ??, zaś wynik takiej dekompozycji (analizy) na rysunku ?. W obrazach detali nie jest widoczna zamglona dolina z uwagi na swój niskoczęstotliwościowy charakter. Liczba współczynników opisujących obraz nie zmienia się - kompresja realizowana jest przy pomocy kwantyzacji, podobnie jak w formacie JPEG. Dokładny opis algorytmu kompresji opisuje Jin Li [?].

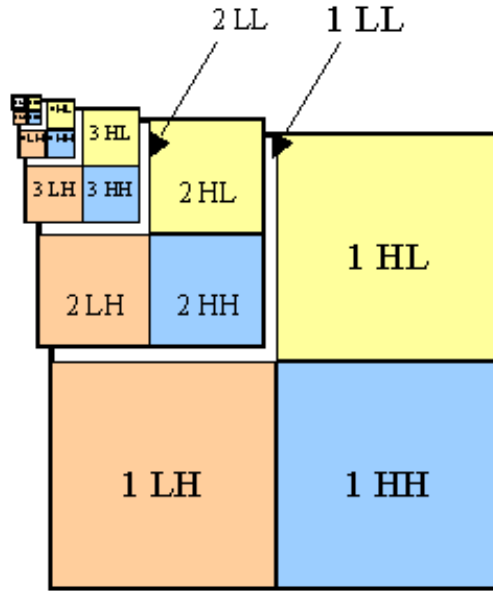
Wybór falki stanowi o wartościach współczynników filtrów G i H. Najczęściej używana jest falka CDF 5/7 (Cohen Daubechies Faveau) dla bezstratnej kompresji lub CDF 7/9 dla stratnej.

Użycie falek biortogonalnych ma tę zaletę że można dokonywać kompresji i dekompresji przy użyciu tych samych filtrów.

W kwestii jakości kompresji oddajmy głos Jinowi Li [?]. Omawiany przez niego wykres 22 to rysunek ??



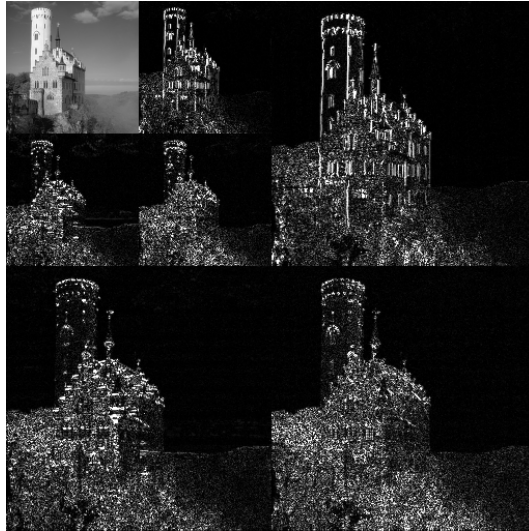
Rysunek 10: Dwuwymiarowa transformata falkowa [?].



Rysunek 11: Współczynniki rozwinięcia falkowego kolejnych stopni. Jak widać nie dają one kompresji i w jednej z wersji algorytmu JPEG (całkowitoliczbowej) są odwracalne.[?].

(14).

We compare the JPEG 2000 image compression standard with the traditional JPEG standard. The test image is the Bike standard image (gray, 2048x2560). Three modes of JPEG 2000 are tested, and compare against two modes of the JPEG standard. The JPEG modes are progressive (P-DCT) and sequential (S-DCT) both with optimized Huffman tables. The JPEG-2000 modes are single layer with the (9,7) wavelet (S-9,7), six layer progressive with the (9,7) wavelet (P6-9,7), and 7 layer progressive with the (3,5) wavelet (P7-3,5). The JPEG-2000 progressive modes have been optimized for 0.0625, 0.125, 0.25, 0.5, 1.0, 2.0 bpp and lossless for the 5x3 wavelet. The JPEG progressive mode uses a combination of spectral refinement and successive approximation. We shown the performance comparison in Figure 22. The JPEG-2000 results are significantly better than the JPEG results for all modes and all bitrates on this image. Typically JPEG-2000 provides only a few dB improvement from 0.5 to 1.0 bpp but substantial improvement below 0.25 bpp and above 1.5 bpp. Also, JPEG-2000 achieves scalability at almost no additional cost. The progressive performance is almost as good as the single layer JPEG-2000 without the progressive capability. The slight difference is due solely to the increased signaling cost for the additional layers (which changes the packet headers). It is possible to provide “generic rate scalability” by using upwards of fifty layers. In this case the



Rysunek 12: Współczynniki rozwinięcia falkowego po I stopniu dwuwymiarowej transformaty falkowej. [?].

“scallops” in the progressive curve disappear, but the overhead may be slightly increased.

Algorytm progresywny o którym pisze Jin Li opisany jest na przykład u Zielińskiego [?] (rozdz. 22.4.2).

## Wykonanie ćwiczenia

Część I JPEG Do wykonania ćwiczenia użyjemy pakietu JPEGtool, autorstwa Darrela Hankersona i Grega A. Harrisa, dostępnego w sieci pod adresem: <http://www.dms.auburn.edu/compression>.

Dla obrazu lena.pgm (pgm=portable greymap) należy wyświetlić jako kolejne obrazki.

1. Obraz oryginalny
2. Wynik działania DCT na obrazie
3. Wynik działania standardowej kwantyzacji.
4. Wynik działania dekwantyzacji
5. Wynik odwrotnej DCT

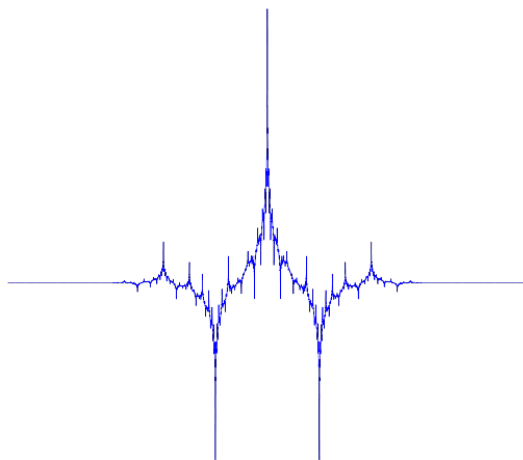
Następnie należy powyższy zestaw powtórzyć dla:

1. Innego rozmiaru bloku (standardowy to  $N \times N$ , gdzie  $N = 8$ )
2. Innego kwantyzatora.

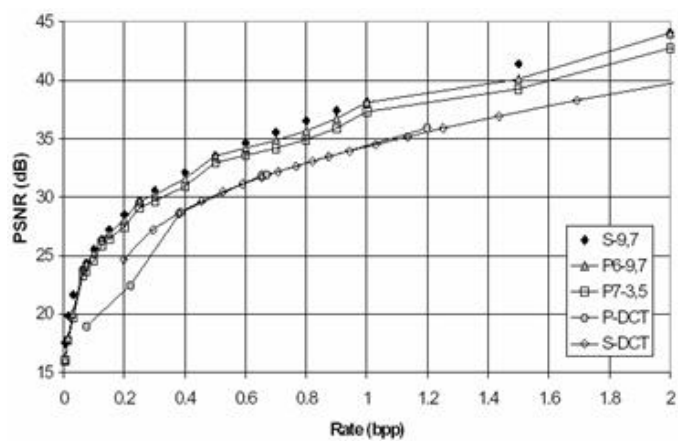
Część II JPEG 2000 (zadanie dodatkowe, dla odważnych)

Do wykonania ćwiczenia użyjemy pakietu jpg2000, autorstwa Arno Swarta, dostępnego w sieci pod adresem: <http://www.math.uu.nl/~sleij101/Opgaven/FourWav/ExercisesWavelets/>. Wymaga on także pakietu huffman dokonującego kompresji, który można pobrać tu: [http://www.mathworks.com/matlabcentral/fileexchange/2818-huffman-coding-and-arithmetic-coding/all\\_files](http://www.mathworks.com/matlabcentral/fileexchange/2818-huffman-coding-and-arithmetic-coding/all_files) albo tu: <http://www.ux.uis.no/~karlsk/proj99/index.html> Zadanie polega na dopasowaniu pakietu jpg2000 (dostosowanego do toolboxa wavelet z Matlabu) do pakietu falkowego WaveLab850 z Octave. Najbardziej istotna kwestia to wpasowanie współczynników transformaty falkowej w odpowiednie miejsca w przetwarzanym obrazie. Resztę odniesień do toolboxa wavelet można inteligentnie pominąć.

**Środowisko sugerowane:** *Octave*.



Rysunek 13: Falka CDF 5/7 [?].



Rysunek 14: Performance comparison: JPEG 2000 versus JPEG (courtesy of Prof. Marcellin, et. al, [?]). [?].

# Bibliografia

- [1] T.Zieliński, *Cyfrowe przetwarzanie sygnałów, od teorii do zastosowań*, WKŁ Warszawa 2009.
- [2] [http://pl.wikipedia.org/wiki/PSNR\\_\(Peak\\_Signal-to-Noise\\_Ratio\)](http://pl.wikipedia.org/wiki/PSNR_(Peak_Signal-to-Noise_Ratio))
- [3] <http://www.mathworks.com/matlabcentral/fileexchange/135-psnr>
- [4] <http://www.answers.com/topic/jpeg-2000>
- [5] <http://www.design-reuse.com/articles/6691/ip-cores-for-accelerating-jpeg2000.html>
- [6] [http://en.wikipedia.org/wiki/File:Jpeg2000\\_2-level\\_wavelet\\_transform-lichtenstein.png](http://en.wikipedia.org/wiki/File:Jpeg2000_2-level_wavelet_transform-lichtenstein.png)
- [7] [http://research.microsoft.com/en-us/um/people/jinl/paper\\_2002/msri\\_jpeg\\_files/image102.jpg](http://research.microsoft.com/en-us/um/people/jinl/paper_2002/msri_jpeg_files/image102.jpg)
- [8] Jin Li, Image Compression - the Mechanics of the JPEG 2000 [http://research.microsoft.com/en-us/um/people/jinl/paper\\_2002/msri\\_jpeg.htm](http://research.microsoft.com/en-us/um/people/jinl/paper_2002/msri_jpeg.htm)
- [9] M. W. Marcellin, M. Gormish, A. Bilgin, M. P. Boliek, textitAn Overview of JPEG2000, Proc. of the Data Compression Conference, Snowbird, Utah, March 2000, pp. 523-544.